Technical Report No. 2

Computer Intelligent Processing Technologies (CIPTs)

Tools for Analysing Environmental Data

Prepared by: Earth Observation Sciences Ltd
Broadmede, Farnham Business Park
Farnham, GU9 8QT, UK

January 1998

European Environment Agency

Cover design: Rolf Kuchling, EEA

Legal notice

The contents of this report do not necessarily reflect the official opinion of the European Commission or other European Communities institutions. Neither the European Environment Agency nor any person or company acting on the behalf of the Agency is responsible for the use that may be made of the information contained in this report.

A great deal of additional information on the European Union is available on the Internet.
It can be accessed through the Europa server (http://europa.eu.int)

Cataloguing data can be found a the end of this publication

Luxembourg: Office for Official Publications of the European Communities, 1998

ISBN

Printed in

Printed on recycled and chlorine-free bleached paper

European Environment Agency
Kongens Nytorv 6
DK-1050 Copenhagen K
Denmark
Tel: +45 33 36 71 00
Fax: +45 33 36 71 99
E-mail: eea@eea.eu.int

TABLE OF CONTENTS

ACKNOWLEDGEMENTS

Contributions

The manual was prepared by staff of Earth Observation Sciences. The chief author was George Simpson with contributions from Pam Vass and Graham. Project management and coordination was provided by Paolo G. Meozzi of the European Environment Agency The assistance of the following individuals is gratefully acknowledged:

- Tim Lack, Water Research Centre, Wallingford – consultancy on applications and technical review
- Howard Grubb, University of Reading – consultancy on data analysis;
- Martin Brown, University of Southampton – consultancy on neural networks and expert systems;
- Grenville Croll, Eastern Software Publishing – consultancy on optimisation and risk management;
- Scientific Computers Ltd – neural network product demos, figures and text;
- staff at the European Environment Agency, and;
- Gurbax Bajwa – product research.

Product information has been gathered from web-sites, advertising literature, news-groups, demos, manuals and direct use of the software. All copyrights are acknowledged.

Trade Marks and Registration Marks

To aid clarity in this document, the Trade Mark and Registration Mark symbols have only been used to identify the products reviewed and listed. However, all products and name brands are trademarks of their respective owners. This includes:
Access, dBase, DG (Data General), DOS, Excel, HP (Hewlett Package), IBM, IDL, Informix, Lotus, Macintosh, Microsoft, NAG, OpenVMS, Oracle, Quatro-Pro, SGI, SUNOS, SUNSolaris, Sybase, UNIX, VMS and Windows.

FOREWORD

As Europe interest in environmental issues increases so does the growth of data that is collected in order to monitor and assess the state and trends of a variety of factors and conditions. The European Environment Agency (EEA) - similarly to many other institutions and professionals - is confronted with an increasing need to extract relevant information from vast amounts of data. The importance of progressing from data to information is nowhere more evident than in the field of environmental sciences where we are witnessing unprecedented data growth.

One way the EEA can boost progress towards this aim is to provide a solid overview and description of the tools that can make data analysis more precise and effective. This is an important issue because, as in so many other aspects of late 20 th century life, we are in an age of rapidly expanding choices. In such an environment it is very difficult to be sure what the best choice is. This is especially the case in the field of computer technology.

Of course, we cannot select technology for people. But we can offer timely information about what is available - including an assessment of benefits and shortcomings - and let them make the decision. To that end, the EEA commissioned the preparation of this report to audit and appraise data analysis technologies with particular emphasis on environmental applications. We trust that it will serve as a useful tool to help environmental workers select the most appropriate information processing technology for their tasks. We ask our readers to assist us in keeping the report up-to-date, by informing us about technologies and products they would like to see featured in future editions.

The EEA is developing with the support of external experts, in many cases the EEA's European Topic Centres (ETCs), reports reviewing the state of the art in relation to constraints or tactics relevant for producing or processing environmental data and information. This manual is a significant building block towards the identification of automated means for improving our data processing needs.

Domingo Jiménez-Beltrán
EEA Executive Director

1. SUMMARY

‘CIPTs: Tools for Analysing Environmental Data’ is a manual to help environmental analysts select the most appropriate information processing technology for their tasks.

Environmental workers face rapidly increasing data volumes, and they need to process these data in the most accurate and efficient way, so that problems can be quickly identified, alternative solutions can be specified, and the state and trends of the environment can be understood. Delivering the highest quality analysis in an understandable form is particularly important when it comes to communicating with the public and lawmakers. New technologies that support these goals are available, but until now they have been accessible only to specialists. The variety of technologies, arcane language, the number of product options and the cost and risk associated with adoption of new technologies have all been barriers restricting their use in environmental analyses.

To help overcome these barriers, this manual offers a guide to advanced information processing technologies, collectively called CIPTs, standing for ‘Computer Intelligent Processing Technologies’.

The manual treats six CIPTs:

1. data visualisation
2. data and trend analysis
3. neural networks
4. expert systems
5. optimisation and risk management
6. intelligent agents.

This order represents the current take-up of the technology by environmental researchers. Data visualisation and data and trend analysis are well used in the discipline, with numerous applications examples. In contrast, although neural networks and expert systems have been around for some time their general use in the environmental sciences is only recently becoming more widespread. Again, this is the case for optimisation and risk management technology, which is only just beginning to be used in process-oriented aspects of environmental monitoring. Finally comes intelligent agents, a new technology for which operational applications are yet to emerge.

Each chapter contains:

- an introduction to the topic, exploring the technology’s capabilities and limitations
- reviews of one or more products that deliver the technology
- a table listing other products that offer convenient access to the technology
- examples of use of the technology in a selection of environmental applications
- guidance to material in which the reader may find more information about the topic.

The manual has not been written in a style that requires you to read it from beginning to end. There is some cross-referencing between chapters, but in general each chapter can be treated as a standalone description of the technology. We advise you to first read Chapter 1, but after that dip into the relevant chapters of interest to you. In addition, each chapter contains a general description of the relevant techniques; this section is intended as a brief introduction and can be skipped by those with a basic familiarity of the technology.

The document is aimed at three kinds of readers:

- environmental analysts: people with responsibility for abstracting information from numerical data, particularly datasets of significant size

- managers of environmental projects: people with responsibility for delivering appropriately grounded conclusions from environmental studies
- environmental scientists: people seeking new ways to make better use of environmental data.

The manual was prepared by Earth Observation Sciences for the ‘Analysis and Exploitation of Existing Information’ section of the European Environment Agency, under project manager Paolo G. Meozzi.

2. INTRODUCTION

2.1. Document Purpose and Scope

A diverse range of organisations and individuals need to analyse data on the environment and make decisions on management or monitoring issues. Numerous technologies exist to assist this process. However, for several reasons., selecting the best technology may not be easy. There are two levels to the problem: the technology itself (usually a mathematical/engineering topic such as neural networks) and the products that make the technology accessible. The first problem is that there are many new technology topics, and most of them are rapidly evolving, which makes it difficult for non-computer specialists to keep abreast. Then, having identified a technology that seems likely to be useful, the next problem is the variety of products that claim to support the different technologies – which one is best for a particular job? Finally there is the cost and risk of applying new technology – it takes money and time, and how can one be sure that the investment will deliver the desired results? Therefore, this manual is designed to help ease the problems by offering a guide to the selection and application of relevant Computer Intelligent Processing Technologies (CIPTs). It treats the following topics:

- data visualisation
- data and trend analysis
- neural networks
- expert systems
- optimisation and risk management
- intelligent agents.

Of course there may be other computer technologies useful in a particular domain; we do not claim to cover all the possibilities. The topics selected are those thought most likely to be useful in the gathering, analysis, interpretation, and application of environmental data. Our terminology 'Computer Intelligent Processing Technologies' or 'CIPTs' needs explanation. Computers are not intelligent and the weight of current opinion is that computers anywhere near as intelligent as humans are far, far in the future. So the title of this manual, while in line with current jargon, could be misunderstood. At their best, what CIPTs do is amplify the intelligence of the people using them. They empower people to make better use of their data by giving them tools crafted by masters. It still is, and probably always will be, up to people to use the tools in an intelligent way. Nevertheless, there is a need to refer to the technologies collectively, hence we will call them CIPTs.

Readers should be aware that there are three boundaries on the scope of this document:

1. Geographic Information Systems (GIS), which is indeed a powerful CIPT, are not covered.
2. The manual concentrates on measurement data[1] , as opposed to simulation data[2]. There are several reasons for this: (a) the significant differences in the nature of the data – gridding and noise/error aspects, (b) the special-purpose tools that work with the simulation packages, and (c) the relatively higher level of computer expertise of those involved in working with simulations.
3. The market coverage is incomplete. This is both of necessity – since the market is large and rapidly changing – and by design. The purposes of the sections on products are to show the technology in action and to illustrate something of the variety of forms in which it is offered.

[1] Data gathered by instruments or observers

[2] Data generated by computer models

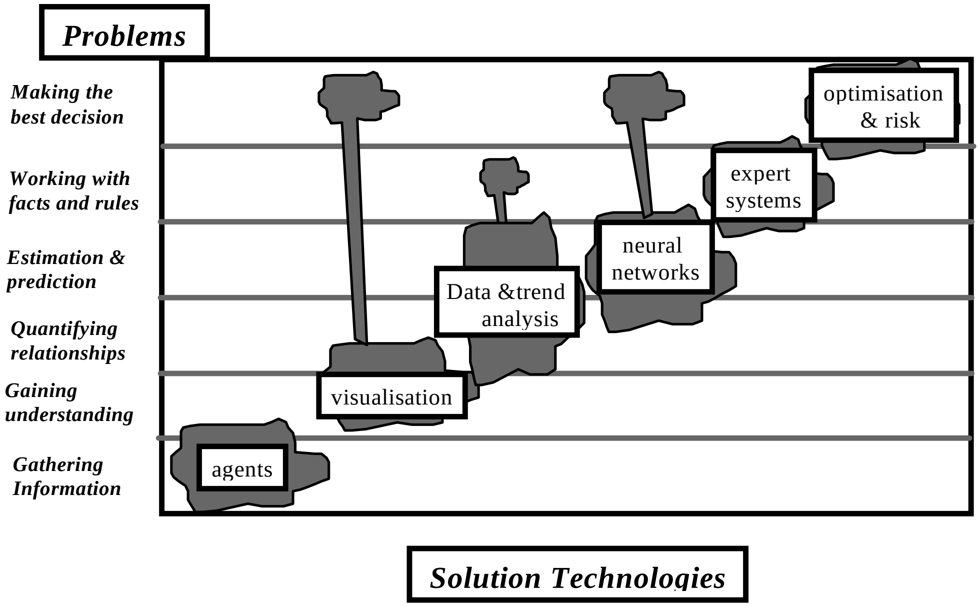
Please keep in mind the fact that our featuring a product in the manual does not mean that we recommend it above others or that it is necessarily 'the current best' – there is no absolute 'best', only 'best for your needs'.

The CIPTs run on some or all of the following machine types: PC, Macintosh, or workstation. It is assumed you are familiar with the operation of your selected machine type (i.e. no guidance is provided in this manual) but you need not have programming experience. Similarly, you do not need an advanced background in mathematics or computing, but you should have some familiarity with these topics. Frequently, the manual refers to material available through the World Wide Web (WWW) – if you are unfamiliar with this technology you should refer to one of the many books on the topic. Finally, you should take care in using the information we present. Although for most of the topics the basic information covered is fairly stable, change is the norm. In particular, new products appear, existing ones are enhanced, and from time to time, some disappear. On a slower time scale, the kinds of solutions evolve. The information in this manual will age, so check the date of issue and do not rely on this document as your only information source.

2.2. Roadmap

To help you quickly find the information you need Figure 2.1 shows six different categories of problems. Corresponding to each of these, the figure indicates one or more solution technologies. As the figure illustrates, some solution technologies are useful for more than one problem. A chapter of the manual treats each of the solution technologies.

Figure 2.1: Environmental problems and relevant CIPTs



The problem categories on the left side of Figure 2.1 represent a logical sequence:

- 'Gathering information' is about using advanced WWW tools to ensure that you know what data and papers are being published on topics in your area.
- 'Gaining understanding' is about discovering what are the important factors in your data.
- 'Quantifying relationships' is about calculating numbers that tell you how important the various factors are in your problem.
- 'Estimation and prediction' is about using the information in your data to find useful numbers, such as an average and its uncertainty, or a best-case and worst-case forecast.

- ‘Working with facts and rules’ is about reasoning with the information in your data, using precise or vague knowledge. For example situations where interpretation of both data and law are required.
- ‘Making the best decision’ is about choosing the best alternative based on all of the available information. For example selecting an industrial development site that minimises environmental damage while still being economically viable.

In each of Chapters 3-8 you will find:

- a summary of the essential concepts
- a frank discussion of the current limitations
- an analysis of the likely evolution of the CIPT
- technical descriptions of one or more products
- a qualitative estimate of the cost factors associated with each of the featured products
- a table listing a wider sample of relevant products
- examples of current environmental applications
- a listing of references and other sources of background information.

For the listings of each product we provided the following:

- name
- brief overview
- product features
- any data limits (if known)
- operating systems – which ones it runs on
- ease of use comments
- other comments (especially supplier, including a web-site address).

It may be that one or more of the CIPTs are already available at your site; if so you may have in-house expertise to assist you in getting started. However, the most important thing is to match the package to the job at hand as they differ widely in their approach, capacity, and ease of use. Please keep in mind that the information we provide is not a complete market survey.

The basis of the material in the manual, from which you may judge its reliability, are listed below:

- each chapter was prepared in consultation with one or more domain experts, who contributed significantly to the material provided and reviewed the finished chapter. (An exception is Chapter 8, for which a different approach was taken).
- the examples of current applications were found by searching the WWW and library references.
- the ‘likely evolution’ sections are the authors’ opinion[3] based on many years experience in science and the information industry, as well as consultation with other experts.
- the product descriptions are based on manufacturer’s information, supported in most cases by actual experience with the product and/or consultation with the vendor.
- the estimates of indirect cost factors will vary depending on the technical skill and experience of the people involved.
- all the product descriptions were gathered during 1997. As prices vary depending on product versions, platforms, discounts, licensing arrangements, etc., we have avoided giving detailed information, which would quickly get out-of-date. For the featured products we have indicated the relative price range of the products (see also Section 2.9) and for all products listed we have provided a web-site address, so that you can obtain the latest information for yourself.

[3] The subjectivity of these sections is made clear by the phrase “The authors’ view…”

- The material listed in the references and bibliography section is a good starting point for learning more, but is not a complete literature review.

The next six sections survey the problem categories and solution technologies in a little more detail. After reading this material you may want to go directly to the CIPTs chapters you think likely to be useful in your domain.

2.3. Gathering Information: Intelligent Agents

You are probably familiar with the World-Wide-Web (WWW) and the convenient tools that allow you to search the immense volume of text that is available through it. If not, you soon will be, because, ‘the web’ is as profound a revolution as the printing press. CERN invented the WWW only a decade ago, and in that brief period of time the web has become the most important global medium for finding information and making information and data available to others.

There are fairly convenient tools, such as browsers, search engines, newsgroups, mailing lists, and Frequently Asked Questions (FAQs) that are very helpful for finding information quickly on the WWW. But when you need to follow ongoing developments in a subject, as opposed to looking into it just once, or you need to do an in-depth analysis, as opposed to finding a sample of all the information that is available, intelligent agents can help by searching and monitoring the web for you. Intelligent agents are software programs that roam the web searching for information that you want. Using agent management tools, you train and maintain your agents to find the kind of information you need.

Personal agents are just the beginning. Powerful distributed information management tools are being developed using agent technologies. These tools are suitable not just for text information, but also for numeric and graphical data. But agent technology is not yet mature enough for most users. For this reason, we have placed the chapter on intelligent agents at the end of this manual.

2.4. Gaining Understanding: Visualisation

Unless your data are very simple, there will be many things you need to understand about it that you can only find out by visualisation (see Chapter 3). A great deal of time can be saved by looking at your data graphically, so that you understand the kinds of relationships it holds, before unleashing the powerful quantitative tools we describe later. One reason for this is the old computer adage ‘GIGO’ – garbage in, garbage out. In other words, if your data are faulty, then sophisticated analyses are a waste of time. Another reason is that data visualisation may tell you directly much of what you need to know, eliminating the need for more sophisticated tools.

The best visualisation tools not only help you discover the relationships in your data, but also let you derive statistics. In some cases visualisation may be all you need to optimise a decision. There is another reason you should visualise your data: it is dangerous not to! All of the more ‘advanced’ technologies can lead you into serious trouble if you do not understand your data. Questions such as the following are quickly answered with visualisation, but can be hard to resolve any other way:

- Is your data clean – free from outliers and dropouts that could bias statistics?
- What is the nature of the relationships – are they linear or non-linear?
- Are the data evenly distributed in the parameters, or are they clustered?
- Do the clusters overlap or are they cleanly separated?
- Are the clusters more or less gaussian in shape, or not?
- What other relationships among the quantities are there?

Visualisation is essential and yet it is limited; the best tools can display a little more than five dimensions. So for datasets with many variables, visualisation consists of plotting all sorts of variable combinations. With good visualisation tools, creating plots takes only a few mouse clicks.

Statistical tools and neural networks are other ways to find relationships in data, and these work even when the data contain too many dimensions to analyse visually. However, it almost always pays to visualise the results of these analyses to cross-check them. Gremlins lurk in unexamined data!

2.5. Data and Trend Analysis: Quantifying Relationships

Having seen for yourself the relationships in your data, the next step is to examine these relationships quantitatively. Environmental data reflect complex relationships, and the powerful data and trend analysis capabilities offered by statistical tools can help in understanding these relationships.

An example: levels of a toxic substance are periodically elevated in a certain waterway. The discharge times and volumes of all of the factories along the waterway are known. Do the data indicate a single source, and if so, how confident can we be of the connection between a particular source and the elevated toxic levels? Properly used statistical tools can allow us to make statements like "based on the data there is a 98.3% probability that the source of the toxicity was factory X, and a 1.7% chance that it came from one of the other factories.

A word of caution: the newspapers hold countless examples of meaningless or misleading statistics. The manual (see Chapter 4) tries to help you avoid some of the more common hazards.

2.6. Estimation and Prediction: Neural Networks

Neural networks (see Chapter 5) are computer programs that are modelled on ideas about how brains work. A common feature is that they learn from experience, usually by slowly adjusting some internal constants in response to a large number of data examples. There is a sense in which neural networks are just a different kind of statistics. The following paragraphs try to explain the difference.

Perhaps the most dangerous, yet also the most successful, assumption underlying much of classical statistics is linearity: that the relationships in your data are simple, for example linear. Linear relationships can be expressed as:

*output = baseline + constant * input*

You may need to make estimates and predictions from you data, even if the relationships are not at all linear. You may be in the position of having no idea about the relationships. You might prefer not to have to be concerned with linearity and non-linearity. In such cases, neural networks are a popular answer. There is a big industry in neural networks for financial data – predicting the stock market, identifying credit risks, targeting advertising campaigns. And there are major applications in signal and image processing. You can think of neural networks like obedient pets – you show them by example what to do, and they do it again. Here are some other aspects that support this analogy:

- they sometimes exhibit inappropriate behaviour
- they may require patient training
- there are lots of different kinds
- some are more clever than others, but they don't reason
- what is going on inside them is difficult to know
- there are lots of 'tricks of the trade' that are useful in getting them to behave.

There are some very advanced kinds of neural networks, for example to automatically find categories in data or to do speech recognition. But this manual focuses on the basic kinds, those that have the most potential for immediate application to the problems of environmental research and management.

2.7. Working with Facts and Rules: Expert Systems

Complex bodies of knowledge present special problems, unless you have that body of knowledge in your head already. Imagine the following rules:

> *The correlation between nitrogen tax levels and steady-state nitrogen-producing industrial discharges is 55 ecu per percentage point change.*
>
> *If an instrument of type N gives a reading in the range 3% to 6% then the conversion factor to obtain nitrogen is 2.7.*

There are powerful technologies for handling rule-based systems. You can use them to answer questions such as :

> *If the nitrogen tax was increased by 100 ecu, what would be the effect on the average reading of instruments of type N?*

This example is trivial, but expert systems (see Chapter 6) can efficiently handle complex chains of reasoning using a great many rules. A word of caution though: verifying the correctness of large expert systems is extremely difficult. In addition, the rules listed above are ‘crisp’ rules. A rule either applies or it does not. There is another kind of system for handling rules, in which this limitation is relaxed. Systems of these kind are called ‘fuzzy’, because they handle vagueness or ambiguity. Fuzzy systems can be engineered to a high degree of robustness. Fuzzy expert systems have for some time been embedded in devices such as washing machines (with about twenty rules) and recently fuzzy helicopter control systems have been demonstrated. Some of the fuzzy rules for a vehicle controller look like:

> *If the destination is near and the speed is moderate, brake medium.*
>
> *If the destination is near and the speed is high, brake hard.*

The terms ‘near’, ‘moderate’, ‘medium’, ‘high’ and ‘hard’ are vague, each applying to some extent to the situation. Fuzzy systems use a simple algebra to compute an output based on the degree to which each of the rules applies.

Yet another type of expert system deals with probabilistic reasoning or beliefs. Medical diagnosis is a familiar example – as the doctor asks you about your symptoms, a large set of possibilities is being pruned down to a single diagnosis. In the doctor’s memory is a set of rules like:

> *Symptom A suggests condition X (most likely), Y (perhaps), and Z (rare).*
>
> *Symptoms B and C, if present, confirm X.*

Probabilistic expert systems quantify and manipulate networks of evidence, allowing users to enter the observations that are available and observe the effect on the other probabilities.

Some possible applications of expert systems to environmental problems include:

- monitoring sensor data to determine if regulations have been breached (hard expert systems)
- controlling environmental treatment plants (fuzzy expert systems)
- diagnosing the causes of disturbances (probabilistic expert systems).

The chapter on expert systems will introduce you to all three types, so you can decide if and how this technology can help in your problem domain.

2.8. Making the Best Decisions: Optimisation Tools

Environmental decision-making frequently includes balancing conflicting factors; cost to the few and benefit to the many or vice-versa. Although the value judgements involved in these decisions can only be made by people, once these judgements have been made, CIPTs can help in finding an ‘optimum’ solution to the conflicting factors. For example, suppose that elevated levels of a certain pesticide in river water are the cause of a 15% reduction in the fertility of sheep. And suppose further that you know the economic impact of the reduction in sheep fertility is 25 Mecu per annum. You are considering two possible measures to reduce the impact:

- supplying water treatment equipment to farmers in affected areas
- placing a tax on the pesticide.

Let us say that the cost of supplying equipment to farmers is 250 ecu per herd, and there are 1000 affected herds, with a variety of different exposure factors depending on where they are located with respect to the sources. And suppose in addition that we know the annual value of the sales of the pesticide in the study area is 140 Kecu, and a 25% tax on the pesticide is expected to reduce its usage by 50% – shifting users to a different product. The scenario continues: the pesticide is used by bean growers, for each Kecu pesticide purchased and applied, the increased value of the crop harvested is 3 Kecu. A tax could have the side effect of reducing the use of pesticide and in turn, lowering the increased value of the crop harvested by up to 0.6 Kecu per farm. You need to know what measure or mix of measures produces the maximum economic benefit.

The problem just posed has a unique answer that can be found using simple optimisation technology. But what if the problem is more complex? What kinds of problems can the best available technology handle? The chapter on optimisation and risk management (Chapter 7) explores these issues. And as its title indicates, the chapter also treats the related topic of risk management, introducing some of the quantitative tools that can help you make the best decisions in the face of uncertain information.

2.9. Cost of Applying CIPTs

One concern you will have as you consider the technologies in this manual is how much it will cost. The purchase price of the technology is often the least significant element of the cost. Other elements are training time, installation time, development time (if required) and the time to do the work itself. These costs need to be estimated and compared to the benefit expected.

CIPTs can be obtained in several different ways, and the one that is best for you will depend on your problem, your resources, and what is available. Roughly speaking, there are three alternatives:

- Self-contained packages that you can use directly – ‘turnkey’ solutions. You point one of them at your data and results come pouring out. If the tool meets your requirements, this is the best solution.
- Using consultancy (in-house or outside) to assist you in configuring a tool to meet your needs. The consultant is an efficient expert, so when the configuration is finished, you can get on with your job. You get a turnkey solution at the end of this process, but you have to find and pay the consultant. And if your requirements change, a consultant must be there to help you again.
- Investing in a flexible ‘environment’ tool that you can configure as required to solve a wide variety of problems. The learning curve is steeper and longer with this approach, but sometimes it is the most reasonable approach. A spreadsheet is a common example of a flexible environment. As a tool for environmental analysis, a spreadsheet is rather restrictive,

and one has to work quite hard to be sure it is correct. But for small problems, it may be the best option.

This manual tries to help you select the right option for you by estimating the learning time associated with a CIPT, the effort required to install it, and the time to solve a problem once the technology is mastered. Of course, these figures cannot be given exactly – user background, amount of local support, quality of documentation, the nature of the problem, and other factors vary widely. For these reasons a ranking scheme is used, as shown in Table 2.1.

The learning curve refers to the time to acquire the skill needed to apply tool in work and application refers to the time required to perform typical tasks using tool. A product with a better ranking should require less effort than others. However, for the reasons just given, the time you expend in any of the categories may vary considerably from the figures given. In addition we have included a ranking scheme to allow prices to be compared.

As mentioned previously because the price of a product varies according to the version selected, the platform you wish to run it on, combinations of products used, whether your organisation qualifies for discounts or licensing arrangements, etc., we felt it unwise to give an indicative price (which would, of course, get quickly out-of-date), but rather we have provided a relative guide and refer all interested users to the vendors web-sites for definitive details or contact numbers. Some products are in the public domain and in these cases available ‘free’ will be included in the table and where a ‘+’ is indicated against a price category (e.g. E+) it means that the price is towards the upper limit of the specified range.

Table 2.1: Ranking scheme for cost factors

	learning curve	installation time and experience required	application	price
A	<10 minutes	<10 minutes of a non-specialist	<10 minutes	1-300 ecu
B	about an hour	about an hour of a specialist	about an hour	301-500 ecu
C	one to several days	one to several days of a specialist	one to several days	501-1000 ecu
D	several weeks or	several weeks/months of a	several weeks or months	1001-2000 ecu
E				2001-5000 ecu
F				5001-10000 ecu

3. DATA VISUALISATION

3.1. Capabilities and Limitations

What is data visualisation?
Visualisation of data from experiments has been an established technique used in scientific research for many years (Abarbanel et al, 1993; Rosenblum, 1994), but it has been with the use of PCs and the availability of CIPTs that the technique has gained wide usage. As termed today, data visualisation is the use of interactive computer graphics to display numerical information. It can range from a graph showing how a measurement changes over time to an interactive, colour, multi-dimensional animation. There are three different uses for visualisation: (1) investigation, (2) validation and (3) presentation. These three uses have different requirements associated with them. In brief, the broad goal of data visualisation is to assist researchers and others in understanding data obtained by simulations or physical measurement, by enabling them to see patterns and understand any underlying relationships in the data. It acts to complement and amplify existing scientific methods for data comprehension and analysis.

The main advantages of data visualisation are:

- it condenses and combines large multi-parameter data sets into a representation that is easier to interpret
- it permits real-time simulations of displays of models or results of experiments
- it allows greater interaction with a data set – users may, if the software allows, manipulate variable combinations, angle of view and control of time stepping while the simulation is occurring.

As the problems under study and their corresponding data become more complex, the need for more powerful visualisation tools grows. The availability of powerful, user-friendly visualisation packages makes it easy to find an appropriate tool to visualise your data.

Why is it important?
Appropriate data visualisation is an essential component of any task involving environmental data, or indeed any significant data analysis task. The reason visualisation is important is because complex datasets are rich in information and there is no better system for discovering and interpreting the relationships in the data than the human brain itself, particularly the visual system.

Another strong point about visualisation is that it makes very few assumptions about your data. This is important because the main task in many environmental analyses is looking for relationships. Visualisation is a powerful, perhaps the most powerful, tool for data exploration.

3.1.1. Key Capabilities

For investigation and validation – the key visualisation capabilities are:

- **tabularisation** to view raw data in an orderly format
- **histogramming** to analyse frequencies of categories
- **correlation plotting** to identify relationships, trends and classes
- **geographical localisation** of data classes.

The essential tools that address these needs are the scatter plot and the histogram. For presentation, animation, in particular WWW animation, is the key capability. The remaining paragraphs in this section summarise the essence of each of these five key capabilities as they relate to currently available CIPTs.

Tabularisation

Data can be presented in many different ways. The simplest form is a table. Tables are very convenient for data storage as all the information that has been observed can be displayed. Table 3.1 shows a fictitious data set consisting of the outside air temperature at noon compared to average fuel oil consumption per day per household. However, simple tables are not always the best way to interpret and digest information. Instead, data could be grouped together in categories or classes and the number of observations within each category is counted (for example, see Table 3.2 which summarises the full 3-month data set partly shown in Table 3.1). When this information is presented in a tabular format it is known as a frequency distribution. Some of the details of the original data has been lost, but clarity has been gained.

Table 3.1: Table of data

Date	Temperature (C)	Oil consumption (litres)
01-Jan	2.1	8.9
02-Jan	3.5	8.8
03-Jan	3.2	8.9
04-Jan	3.1	8.6
05-Jan	4.7	7.2
06-Jan	5.1	5.2
07-Jan	3.2	7.9
08-Jan	3.1	7.6
09-Jan	1.1	9.0
10-Jan	-0.5	10.2
11-Jan	0.6	10.0
12-Jan	0.1	9.9
13-Jan	0.6	9.8
14-Jan	1.6	7.6
15-Jan	2.1	7.4
16-Jan	2.6	7.3
17-Jan	3.4	6.8
18-Jan	2.4	7.2

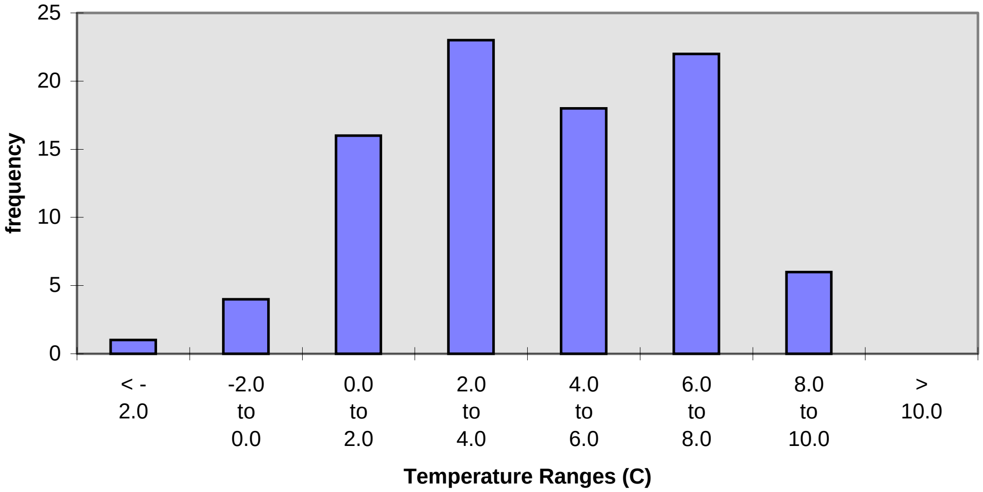
Table 3.2: Frequency distribution for temperature

Ranges in Centigrade	Frequency
< -2.0	1
-2.0 to 0.0	4
0.0 to 2.0	16
2.0 to 4.0	23
4.0 to 6.0	18
6.0 to 8.0	22
8.0 to 10.0	6
> 10.0	0

Histograms

Clarity can be further increased by transforming a frequency distribution into a diagram known as a histogram. As Figure 3.1 illustrates, the basic histogram is a bar chart showing the frequency with which measurements fall into a number of defined ranges.

Figure 3.1: Histogram of temperature



Correlation: scatter plots

If the measured variables can be plotted against each other then the information might be plotted on a graph or scatter plot showing correlation. These give a good visual impression of the strength of the relationship between the variables and may convince an analyst that no further work is required to establish or refute the relationship. Scatter plots can display relationships (Figure 3.2) or trends with time (Figure 3.3). In the latter case, the plots are often referred to as time series. In the analysis of these data, oil consumption will probably be assumed to be dependent on temperature and not vice versa. In many cases, although correlation may be visible, it may still not be clear that there is any dependent relationship or even common causality.

Figure 3.2: Scatter plot showing relationship between consumption and temperature

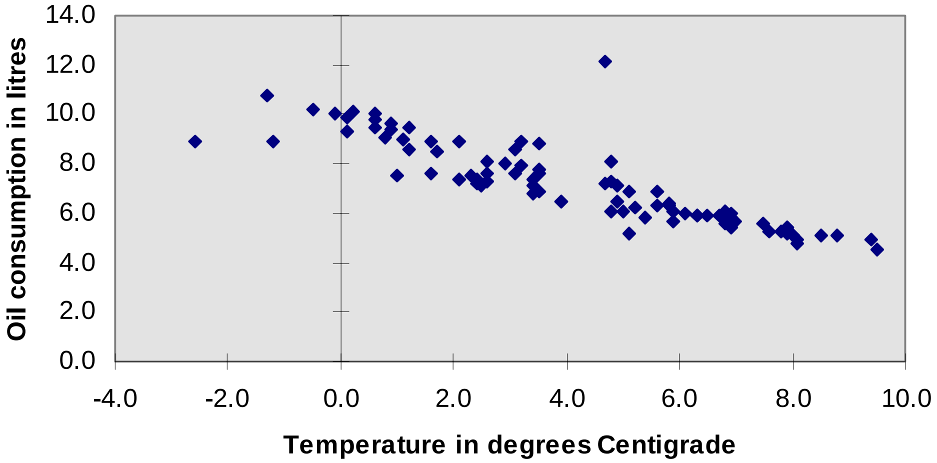
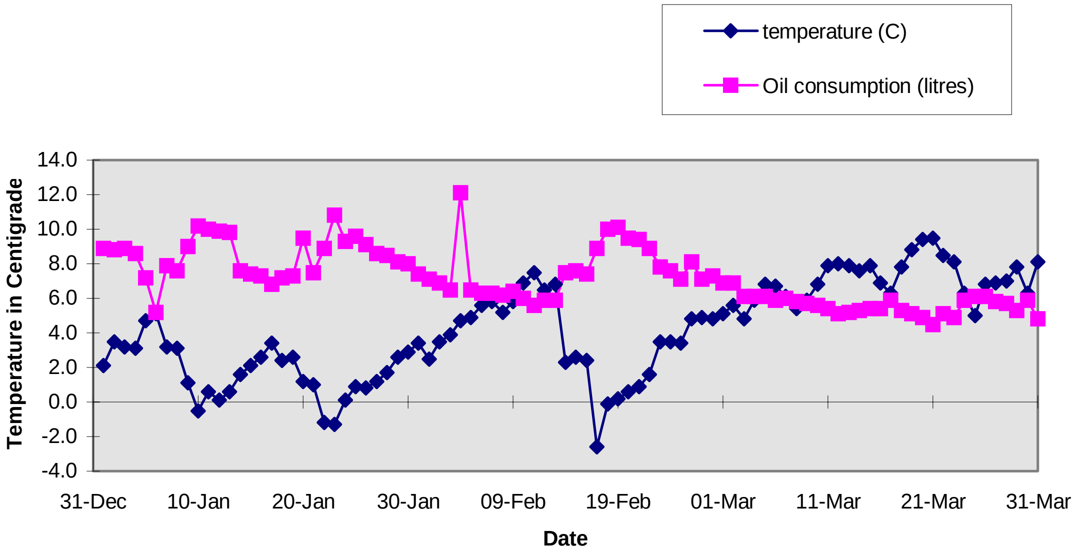
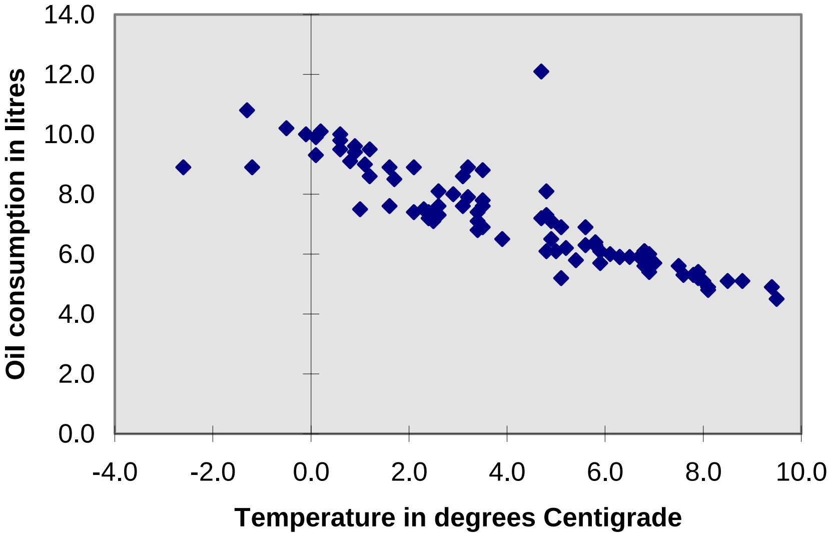


Figure 3.3: Scatter plot showing trends over time for temperature and consumption



Correlation fitting (or regression) is a range of statistical techniques used to determine the strength of relationships between variables. Correlation between variables is often displayed graphically by adding a regression line (or line of best fit) to scatter plots. For example, in the 2-variable scatter plot illustrated in Figure 3.4 each axis is one measurement variable, with each measurement represented as a point in this co-ordinate system. A best fit linear correlation is illustrated as a single straight line. In many situations, 2-D scatter plots will reveal data clusters that correspond to significant phenomena. Simple linear correlation may be inappropriate when more complex phenomena arise (see Chapter 4 on this topic).

Figure 3.4: Correlation fitting



Scatter plots reach a limit of usefulness when there are many variables to consider. There are tools that allow you to plot and view scatter plots in three or even more dimensions. For complex datasets, this capability is extremely valuable.

Correlation and scatter plots with complex data sets

For data in which the variables are inter-related, the basic histogram, 2-variable scatter plot and simple correlation may be inadequate. A 2-variable scatter plot shows qualitatively how the data vary qualitatively, but what is needed for more complex data is the ability to display how much of the data falls in each significant region. Approaches diverge at this point.

One avenue is 2-variable histograms – select any two variables and a plot scale, and the computer will show you how many measurement points fall into each bin of the plane. This could also be described as a 'frequency surface' where a 3-dimensional plot has a surface that varies along a 'height' axis according to frequency and along the other two axes according to the histogram bins of each variable being examined. Figure 3.5 shows an example of a 2- variable histogram. Such displays are useful for quantifying features identified in scatter plots, and are essential when the density of points is so high that scatter plots become saturated. In this dataset, there appears to be two populations. Another useful approach is to transform the co-ordinate axes so that the interesting features do show up in a single variable histogram when they are projected on the new axes. Such transformations may be achieved through the application of Principal Components Analysis that seeks out and creates artificial axes in the data where variability is clearest. This can be useful, but as Figure 3.6 illustrates, it is not guaranteed to show you everything. Histograms of the data along the new axes describe the variability better than the original X-Y axes, but some interesting relationships may still lie hidden.

Figure 3.5: Two-variable histogram

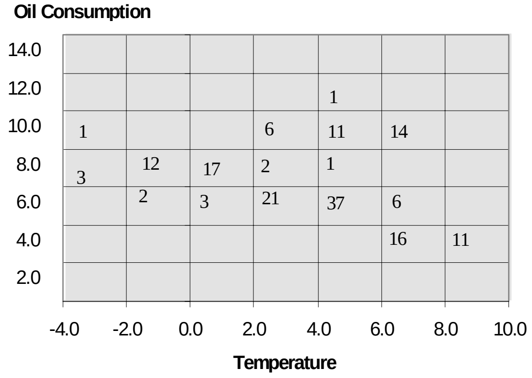
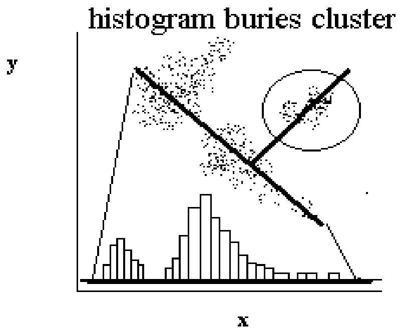


Figure 3.6: Hazards of data projection



A third approach goes back to the scatter plot. Using graphics, significant data clusters are identified interactively, such as in a 3-dimensional viewer that rotates scatter plots using up to 3 axes. This approach is recommended.

Geographic localisation

For analysing the relationship between classes and geographic locations, the ability to portray your data on a map is essential. This is of course a primary function of Geographic Information Systems (GIS), but under certain circumstances the capability can also be realised with standard visualisation tools. If your data is dense and the records in your data have co-ordinate attributes, a colour scatter plot of the co-ordinates with the colour encoding the classes can do the job. Thus for example high levels of radioactivity found in housing can in some cases be correlated with local geology; and this is apparent simply by visual inspection of the localised data.

Similarly, 3-dimensional visualisation of data sets in relation to the environment in which the data were collected can aid in analysis. For example, dramatic changes in soil moisture may be related to the rain shadow effect near local mountains. This would be well illustrated by displaying the mountains in relief with soil moisture sample points located appropriately and examined in relation to elevation and aspect. Another example in the built environment; the location of air pollution measuring devices is critical in relation to roads, buildings, wind direction and aspect. The use of 3-dimensional visualisation tools will help to explain measurement values even before the data are analysed in any statistical sense.

Animation

It is often important to present the significant features of your analysis to a wide audience. The problem here is that the readers or viewers may not be willing to give you much attention; you need to make your point powerfully in a short time – in presentations or on the WWW.

It is now possible for researchers to present their environmental findings as an animated graphic, with sound effects if desired. Essentially the same kinds of technologies used for expensive television advertisements are now widely available. These are suitable for short (~60 second) animation, which can be superbly persuasive (see for example: *http://crusty.er.usgs.gov/*, in which the impact of a new centralised fluid waste discharge system for the Boston area is shown[4].

The major advantage of animation is presentation to non-specialists and the general public. The message goes across quickly and powerfully. The disadvantage however is that the power of the medium can obscure the real meaning of the data, leaving the animation open to charges of bias. The exact form of the presentation becomes critical. For example, isolines of radioactivity shown emanating from an industrial site will be shaded or coloured. Critical thresholds in terms of safe levels could be shown as isolines equal in significance with others or as critical boundaries encompassing ‘blighted’ areas. The danger of subjective interpretation is high. These problems do not generally affect more mundane visualisation techniques.

As animation is slightly outside the realm of ‘Tools for Analysing Environmental Data’, we will not go into this topic further.

3.1.2. Limitations and Likely Evolution – the Authors’ View

What is wrong?

Even with the best of CIPTs, data visualisation is still far more painful than it need be:

- the programs have improved, but they are still inept at handling data that is not in the format they require
- they may be restricted to run on particular platforms

[4] To view this animation, you will need to download the appropriate software. Guidance is provided at the site.

- they have limits on the amount of data that can be displayed and as these limits approach, they slow unacceptably
- some tools require specific programming skills
- the mouse and keyboard are the wrong tools for the job; there is no usage of the other senses – acoustic and kinaesthetic.

In reflecting this, there has been a major focus within the literature on the limitations of visualisation software and tools. In the past, the concern was that they assumed too much knowledge on the part of the user and often did not have the capabilities incorporated within a single package to accomplish many desired tasks. More recently, emphasis has shifted to the absence of visualisation reference models upon which to base systems, tools and visualisations. Another major limitation reported is the lack of consideration of perceptual psychology when creating visualisations. Certain types of presentations may actually be scientifically counter-productive as they use colours, symbols or representations that make it difficult to objectively interpret the data. There is a very large body of work in perceptual psychology that has examined the strengths and weaknesses of the human perceptual system; unfortunately, there are no set rules for applying these results to visualisation. Furthermore, data visualisation can be limited, if it is used without any other data comprehension or analysis techniques. Data visualisation does not provide a complete solution to data interpretation in itself, it only acts as a complement to existing techniques, for example, numerical and statistical methods (Treinish, 1993).

What is happening?

The main energy in computer visualisation technology today focuses on depiction of real-world objects; data visualisation is a sideline. A key issue for real-world visualisation is 'rendering' – making a scene look natural by simulating lighting and shadows. Rendering is a processor-intensive task, but because there is such a strong interest in it, making it fast (and thus making visualisation fast) is an industry priority. Until recently, hardware that supported fast visualisation was confined to specialist vendors – Silicon Graphics built an empire in visualisation. In the past several years, the capability has appeared in most vendors' machines.

Interactive visualisation demands that the data be in the computer's random-access memory (RAM), rather than on disk. So a limit is the amount of RAM available. RAM chip capacity and processor performance continue to double every 18-24 months (even accelerating recently). But there is a snag – memory speed is improving much more slowly than this. There is therefore, a limit to the number of data points that can be visualised simultaneously. Of course, the screen itself sets another limit, not to mention the eye. Release from the tedium of the mouse and keyboard is not far away. Devices that recognise gestures are appearing in the video arcades, and soon will be put to work in business and environmental domains.

Two other recent technical development arenas related to visualisation are VRML and data mining:

- **VRML** – Virtual Reality Markup Language – is a way to create animated 3-dimensional interactive environments. It and similar languages are being used to explore ways in which data can be visualised more effectively (see for example: *http://www.crg.cs.nott.ac.uk/Virtuosi/*).

- **Data mining** – Commercial organisations have discovered that competitive advantage can be gained through analysis of their data. Supermarkets study the buying tempo and tastes of their customers on a national basis to optimise their distribution network. Banks analyse the activity patterns in their accounts to improve their margins (see for example: *http://www.mip.com.au/clem.htm*). Such low-end data mining visualisation packages could well find application for environmental datasets.

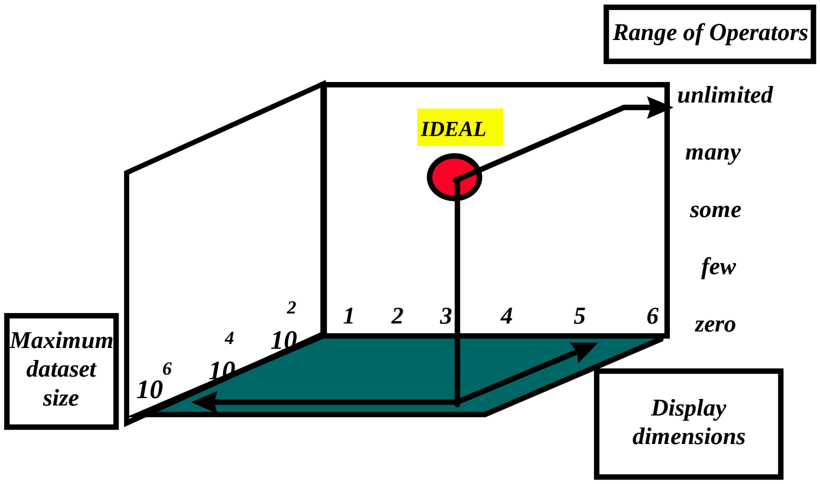
Where is it heading?
Except for animation, the tempo of change in visualisation is not as rapid as we would like to see. The reason for this is not clear. It may have to do with the fact that the majority of problems that businesses need to address (and thus developers are willing to produce solutions for) are not that complex. Another possibility may be that the data have not become that sophisticated. A third possibility is that the most important factors in business situations may be broad and imprecise ones – product appeal, marketing and the like, that are not easily visualised. A fourth is that people may have enough confidence in other CIPTs (such as neural networks) that they do not feel the need to use visualisation. Nonetheless, we can predict with some confidence that with the improvement in communication speeds and the ability of more devices to communicate information, the need for more powerful data visualisation systems will gain urgency and improved solutions will appear. The benefit for environmental users will be more powerful ways to see, and thus understand, the complex and rapidly growing datasets generated by environmental monitoring and modelling.

3.2. Representative Visualisation Tools

Figure 3.7 shows three of the most important factors in selecting an appropriate data visualisation tool for your application. An ‘ideal’ tool would handle as large a dataset as we would like, with the ability to display relationships between as many variables as our brains can accept (which is about six), and which could manipulate the data using an unlimited range of operators. Of course, there are other important factors as well:

- ease of use
- cost and availability
- the different ways the tool lets you display the data

Figure 3.7: Data visualisation tool selection factors



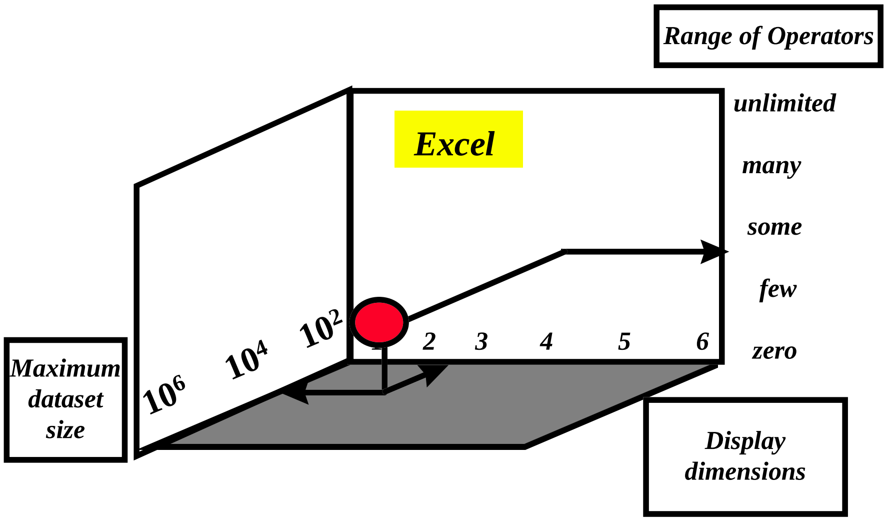
Unfortunately, the ideal tool does not exist. In any case, what you need is not the best tool in an absolute sense, but the best tool for you and your job. We will consider three classes of visualisation tools, showing where their capabilities place them on the axes of the diagram, and discussing the other factors. The three classes we will consider are:

- spreadsheet based tools
- visualisation development environments
- visualisation packages.

3.2.1. Microsoft Excel™ – a spreadsheet visualisation

The example tool we consider in this category is Microsoft Excel. It, like most other spreadsheet applications, offers an associated charting function. As the figure illustrates, the tool has limitations relative to the ideal. Nevertheless, for many environmental data visualisation problems, a spreadsheet solution is quite appropriate. An advantage of this way of working is that the data are dynamically linked to the display, so that immediately after you alter a number in the spreadsheet, the display adjusts accordingly. Many users are already comfortable with spreadsheets, so there may be less of a learning curve. The selection factor ratings for Excel are shown in Figure 3.8 and discussed below.

Figure 3.8: Microsoft Excel™ visualisation selection factor ratings



Range of operators
The range of operators – built in functions – in Excel is quite wide (223 at the last count), but the number of these that are useful for visualisation is rather few. Virtually absent are tools to interact with the data visualisation – selecting and naming subsets of the data is an important example.

Display dimensions
Spreadsheet-based visualisation tools are best for uni-variate data and time series; they typically do not perform well for data with more than two dimensions. A colour scatter plot can be used to display a bit more information, but this is not a very satisfactory solution.

Display modes
Excel offers bar charts, line graphs, pie charts, scatter plots, and various '3D' special effects charts.

Maximum dataset size and dataset type
Excel is useful for small datasets – it does not perform well when there are more than 1000 data points. It handles spreadsheet data only.

Ease of use
Excel visualisation is accessible to any user willing to invest the modest amount of time required to learn to use the tool. The Microsoft manuals and on-line help for Excel are not brilliant; we recommend investing in a good third-party manual.

Cost and availability
Excel is available for PC and Mac platforms, but not for UNIX. Recently however, UNIX platforms have been gaining the ability to run PC applications. Table 3.3 shows the cost factor rankings for Microsoft Excel, using the categories defined in Table 2.1.

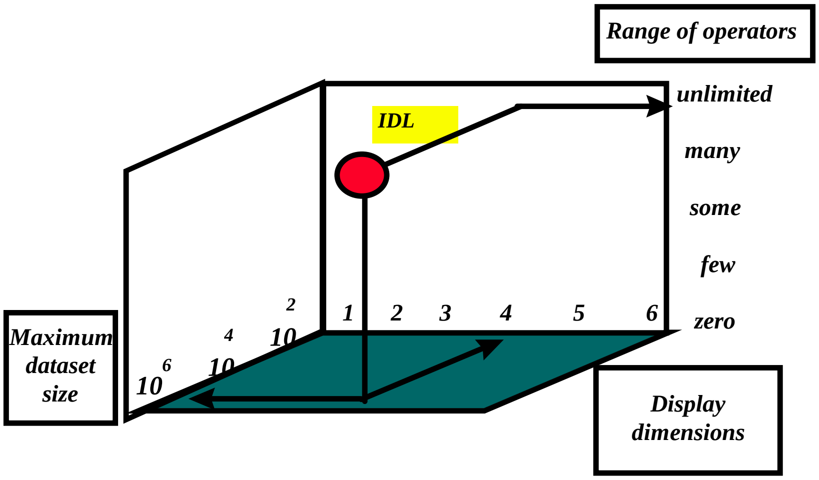
Table 3.3: Cost factor rankings for Microsoft Excel™

learning	installation	application	price
C one to several days	A-B about half-an hour	B about an hour	A

3.2.2. IDL™ – a visualisation development environment

IDL or Interactive Data Language is a high-level interactive programming language for data visualisation. It also supports rapid development of visualisation applications. Its selection factor ratings are illustrated in Figure 3.9.

Figure 3.9: IDL™ visualisation selection factor ratings



Range of operators
IDL offers a wide range of operators and built-in functions. Since it is a programming language, the range of operators is unlimited.

Display dimensions
IDL offers 3-D graphics and data can be plotted in 1, 2, 3 or 4-D. Some graphical forms can be integrated; for example contours can be overlaid on an image.

Display modes
IDL is capable of displaying traditional data representations such as graphs and bar charts as well as displaying images, geometric shapes and solids, free-form drawings, animations or map projections.

Maximum data set size and data type
IDL is optimised for working with large, multi-dimensional data sets. Limits to size of data set are usually a result of hardware limitations rather than IDL imposed restrictions. It handles almost any kind of file; the standard formats are TIFF, GIF, PICT, XWD, SRF, BMP and JPEG.

Ease of use
The basic interface is a command line interpreter, which means that the user types text commands that are executed immediately, rather than having to go through a compile-build-execute cycle. The commands are ‘scriptable’, that is they can be recorded and played back.

Commands can also be executed from a file. The graphical user interface (GUI) toolkit allows IDL programmers to create more user friendly interfaces, which gives the programmer complete control over the design of the visualisation. However this means there is no guidance regarding the most appropriate data representation. Complete control does allow IDL applications to be tailored to meet specific requirements, and is very useful when the requirements change. IDL interfaces with C and FORTRAN and facilitates the exchange of data between both languages. An application does not necessarily have to be built in order for the user to visualise data. Data can be displayed directly from the command line.

Documentation and support

Support for IDL users is available through the IDL Users' Guide, IDL Reference Guide, IDL Mathematics Guide and the Scientific Data Formats Guide. Hypertext on-line manuals and a basic tutorial are available. Some reviews have commented negatively on the documentation. Customer Support Services are available for maintenance, updating, training and consulting. There is an IDL Usenet group.

Learning time

The amount of IDL code needed to do a visualisation task is considerably smaller than with other scientific programming languages. Its command syntax, similar to FORTRAN, makes IDL accessible to scientists with minimal programming experience. Building an IDL application does require programming skills, so that for most users the learning time will be of several weeks.

Cost and availability

IDL is available on all popular platforms (*http://www.rsinc.com*). Table 3.4 shows the cost factor rankings for IDL, using the categories defined in Table 2.1.

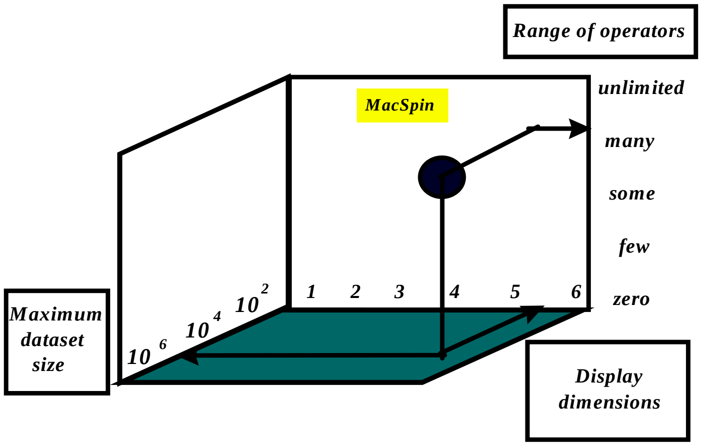
Table 3.4 Cost factor rankings for IDL™

learning	installation	application	price
C-D about a week	B about an hour of a specialist	B-C hours to days	E+

3.2.3. MacSpin™ – a visualisation package

Although it cannot handle the large datasets that IDL can, for the user seeking a visualisation tool that does not require programming, MacSpin is the best we have seen. Designed with the single objective of visualisation in mind, MacSpin is almost guaranteed to deliver surprising insights into your data. We once had the task of looking at several hundred data points, for each of which there were 128 variables. The task was to identify the subset of variables that most strongly discriminated among the classes in the data. Using MacSpin, we did it in a day, including computing some new variables that were combinations of those given. For some users, it will be worth buying a Macintosh just to use it. Spin technology is licensed to a number of other package vendors, so you may be able to get the benefit of its visualisation approach in this way. For example, the SAS statistical package (see Chapter 4) includes a '3-D spinning globe' module. MacSpin's selection factor ratings are illustrated in Figure 3.10.

Figure 3.10: MacSpin™ visualisation selection factor ratings



Range of operators

MacSpin offers the ability to operate on data in two ways – through user-defined variable transformation (like Excel), but more significantly, through set operations on the population. This set facility is uniquely valuable to analysts, so it will be described in some detail. Imagine data containing outliers, that show up as extreme values of random fields; sometimes x is high, sometimes y and sometimes z or alpha. With MacSpin, a subset of the data can be created by setting the display axes to show two or three dimensions. For example, set axes to x and y and select the region of the display that contains outliers. Name the subset 'extremes'. Then from another pair of variables, identify more outliers. Using the 'union' operation, these events can be added to 'extremes'. When all the outliers have been identified, they can be excluded from the display. Alternately, exclude all other data, and investigate the extremes. The other set operation is intersection: given two sets, find the data that belongs to both of them. A lot of interesting analysis can be done with just the union and intersection operations. Finally, MacSpin offers basic statistics on any variable. So for example, it is possible to find the number of extreme members and the distribution quartiles of any variable, as quickly as you can point and click.

Display dimensions

Through the use of animation, colour, symbols and a 'time-slicer', MacSpin can display six different aspects of any data simultaneously. This much information may take some time to comprehend, but for complicated datasets it may be the only way to get the facts out of the data and into your brain.

Maximum dataset size

MacSpin can handle as much data as you have memory. In a previous study a dataset of 12,900 points was worked with intensively, where each data point had 37 variables. This used approximately 7 Mbytes of memory. In the course of the analysis, 44 subsets were identified and 7 new variables created by transforming others.

Display modes

MacSpin's emphasis is on scatter plots – the 3-D colour animated display illustrated in Figure 3.11 being the 'queen' of visualisation. It also supports 2-D scatterplots, bar charts, pie charts, and 1-D histograms.

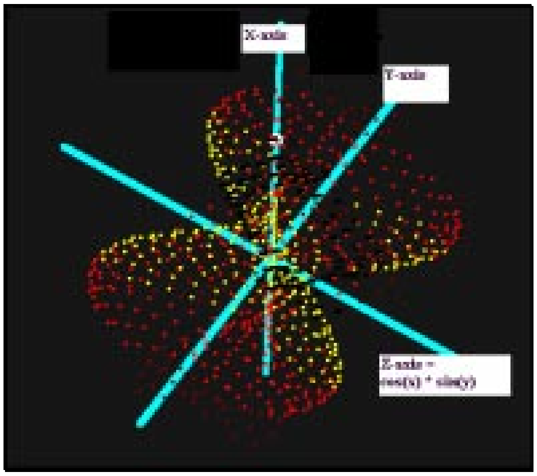
Ease of use

MacSpin has many features and although learning to use any one of these is easy, learning to use them all takes time. One hurdle to cross is getting data into MacSpin[5]. It is advisable to spend half

[5] Save a MacSpin sample dataset as a text file, open it with a spreadsheet such as Excel, and use it as a template to format your data the same way. If necessary, transfer your data to the Mac and open it there with Excel. Add the column label 'Label' above the row identifiers. Then save it as a tab-delimited text file. Open with MacSpin.

a day or so with the manual, playing with the examples and exploring the menus. There are lots of ‘option-click’ and ‘cmd-shift-option click’ features within the menus that can be very useful once the essentials have been covered.

Figure 3.11: MacSpin™ display modes



Cost and availability

MacSpin is available from Donoho Design Group (*http://www.ddg.com/*). Sample datasets are also available. Table 3.5 shows the cost factor rankings for MacSpin, using the categories defined in Table 2.1.

Table 3.5: Cost factor rankings for MacSpin™

learning	installation	application	price
B-C hours to days	A less than 10 minutes	B-C hours to days	free

3.2.4. Other Products

Table 3.6 provides summary details of some of the other visualisation packages that are on the market today.

3.3. Environmental Applications

Visualisation techniques, tools, images and videos abound. But how good are they in environmental applications? How well do they convey the critical information to a user? Case studies have a crucial role in assessing how well particular techniques or approaches achieve the goals in focusing what is needed to improve the effectiveness, reliability and consistency of visualisation across a range of fields (Robertson and Silver, 1995). Two good sources to gain more information about a particular tool of interest or an application area are the individual vendor web-sites, many of which include case study reports, for example Data Explorer, IDL, Khoros, MATLAB, OpenGL, PV-Wave, etc. (see Table 3.6 for web-site addresses), and the journal IEEE Computer Graphics and Applications, which regularly covers visualisation cases studies and the annual IEEE Visualisation Conference. In addition, a number of web-sites are being created to increase visualisation awareness in general (see Introduction to Visualisation: tutorial: (*http://www.cs.uml.edu/~grinstei/tut/v96_tut2.html*) and for researchers involved in particular disciplines, for example a site for bioinformatics and biology can be found at: *http://industry.ebi.ac.uk/~alan/VisSupp/VisAware/*, which catalogues sites, applications, techniques and papers of interest to the community. Furthermore, there are some good overview documents, such as Denzer (1993a), who reviews visualisation-related work on environmental protection in areas of monitoring and control, information retrieval, evaluation and interpretation, and decision support. He describes (see also: *http://www.eas.asu.edu/~voegele/biblio/dagstuhl.html*) requirements with regard to data typing, data processing, user interfaces and graphical presentation of results. These sources have been used to provide some case studies to illustrate the use of visualisation techniques for environmental applications. The small selection included here covers examples in:

- climate modelling and meteorology
- air quality monitoring
- hydrology
- coastal and ocean applications
- geographic information systems
- forest management.

All of the documents cited and other bibliographic references are listed in Section 2.4.

Climate modelling and meteorology

A useful case study report is provided by Treinish (1995) on visualising scattered rainfall data in Peru during the 1982-1983 El Niño phenomenon. He clearly explains the steps in the visualisation process (using Data Explorer) and relates various choices to the interpretation task. The importance of ‘data treatment’ is highlighted, in this case a gridding approach, in analysing the data and understanding the science. Treinish concludes that the techniques described are suitable for other earth sciences, including hydrology samplings. Also related to climate, Max et al (1993) investigated visualisation in climate modelling, focusing on techniques for representing data, e.g. volume rendering, textured contour surfaces and vector field rendering. They wanted to simulate the dynamic Earth’s atmosphere to investigate problems such as global warming and air pollution.

Air quality monitoring

Roth and Guritz (1995) used AVS to visualise volcanic ash clouds from Mount Redoubt in Alaska. Their study treats the integration of several different data sets, including satellite data, to provide a basis for a prediction and decision support system in hazardous situations. The raw data input to AVS were x, y, z co-ordinates for each ash particle. With a particle density array of 150 x 150 x 50 cells, 1.1 million data points resulted for each solution point in time. This data was displayed as an iso-surface giving the viewer a visual effect showing the boundaries of the ash cloud. Plumes generated at five-minute intervals were displayed to create a flip-chart animation, which was subsequently smoothed using linear interpolation. In the future they intend to look at volume rendering to produce more realistic representations of the ash cloud. The authors see their work, also, being applied to the area of pollutant transport modelling and remediation, for example oil spill monitoring. Another air quality application is demonstrated by Wood et al, who used IRIS Explorer to visualise air quality data, collected in real-time from a number of sites around the UK (http://www.nag.co.uk/0/ visual/IE/iecbb/Render/Issue6/VisWeb.html).

Hydrology

Kirsty Tulloch (*http://www.geog.unsw.edu.au/~kirstyt/index.html*) investigated data visualisation as a technique to study water infiltration into soil using IDL. She found that “IDL offers a wide variety of options, thus trying to incorporate within a single package, the capabilities to accomplish all anticipated desired tasks. It is also optimised for working with large, multi-dimensional data sets and reads and writes data easily so that it works with almost any kind of file. Additionally, the amount of IDL code needed is far smaller than that used in other scientific programming languages and its code has a natural command syntax ... and [is] thus more user friendly to a scientist that may have had no experience in the programming area”. However, she felt her work was limited by choosing one type of visualisation from the possibilities offered by the tool. A more appropriate approach would have been to design a variety of types of visualisations using a number of tools, and then select the one that was the most appropriate.

River basins play a key role in the hydrologic cycle, but the precise dynamics of these systems are still poorly understood. To improve this situation, hydrologists first need a way to measure and visualise the intricate 3-D geometry of river basins. Traditionally, hydrologists spent many tedious hours making measurements by hand from topographic maps, which precludes the possibility of studying large basins. Scott Peckham (*http://cires.colorado.edu/people/peckham.scott/RT.html*) frustrated by the limitations of existing C and FORTRAN code developed an integrated 'point-and-click' toolkit (RiverTools). He chose to use IDL, for reasons of ease of use, extendibility, speed and platform-independence. The sole data input to RiverTools is a raster grid of elevation values, a DEM. Users generate a digital map of the river networks in the region, which is stored in a vector-based 'tree' data structure. More than a dozen parameters are automatically measured for every node in the tree, including geometric characteristics like upstream area, channel lengths, drops and slopes. Many topological aspects, such as Shreve magnitude and Horton-Strahler order also are measured and allow the internal structure of the river basin to be described in precise terms.

Coastal and ocean applications

King et al (1995) discuss the use of readily available visualisation and animation software (such as IBM's hardware-independent Data Explorer) to improve understanding of coastal and estuarine current flow dynamics and model results. Several animation techniques are discussed in the paper for application to estuaries and coastal water bodies at grid sizes of 200 m to 2000 m. They looked at an estuarine system, a coastal system and a continental shelf system, beginning each visualisation with a 3-D view of the bathymetry of the region, in order to clearly show the nature of the environment in which the water flows. For the simulated coastal and estuarine flows, patchiness of the current field in both time and space was clearly shown using animation and colour overlays. The patchiness was exaggerated by choosing a colour map with a skewed distribution, as the distribution of the speeds from the model results was skewed in these examples. The hydrodynamic model results (an Eulerian approach) alone did not supply the non-technical viewer with the important information needed to assess the transport characteristics of the flows. However, using a Lagrangian approach, a dye trajectory animation clearly showed the true tidal excursions, the net along-shelf drift, the residence times of water and the mixing rates over transects of the region. By showing both approaches the viewer readily observes the details of the current structure and can realistically visualise how the current field influences the transport of water and material in the region. For further details see: *http://ibm590.aims.gov.au/reports/visual/visual6.html.*

Also relevant to coastal environmental monitoring, Waisel (1996) describes the use of Data Explorer to create 2-D and 3-D visualisations of the sediment chemistry in New York's harbour, where high concentrations of sediment contaminants prevent regular dredging maintenance of the shipping channels, thus adversely affecting the economic viability of the port. The visualisations show the probability of adverse biological effects in a given area, based on both sediment chemistry and sediment toxicity data and locations which exceed reference values for any contaminant or a class of chemicals *(http://www.rpi.edu/locker/69/000469/cv/sedvis/sedvis.html).* Problems were experienced with scaling and the translation of the 3-D representation to a 2-D top-down view following rotation – both problems had to be overcome by customised programming.

Away from the coast, Johannsen and Moorhead (1995) describe the visualisation of ocean flow models, discuss the data handling requirements of large data sets and introduce a technique for depicting vector magnitude and direction by using colour saturation and hue. They make a valuable point about the risk of misinterpreting a visualisation because of implicit but incorrect assumptions. In this case they warn against interpreting the animated colour representations of eddies as if they were dye traces; because dye tracing is so strongly etched into an oceanographer's working paradigm that it may be difficult to override the implicit interpretation mechanism simulated by a visualisation.

Geographic information systems

To overcome the data visualisation limitations of most GIS systems, Eva-Marie Stephan, working at the University of Zurich's GIS lab *(http://www.geo.unizh.ch/~stephan/)*, used IDL to design a visualisation environment to provide interactive viewing and exploratory data analysis capabilities for validation of complex data sets, including spatial interpolation, detection of outliers and false model assumptions of Arc/Info data. Stephan reports that "IDL is a very good environment for easy and complex GUI building. The availability of a variety of interface controls and graphical routines, and the possibility to add new routines using a powerful programming language, is a very good concept. "When I prototyped the program I began with standard IDL routines. But, as I became more familiar, I wrote my own. There was a smooth transition, which I appreciated". Stephan developed DataScaping on a Sun SPARCstation, but has moved the code to a PowerMac and Silicon Graphics Onyx, where it ran "without any significant problems or code changes".

Forest management

Brian Orland *(http://imlab9.landarch.uiuc.edu/SF/SF_II.html)* describes three development projects in forestry addressing data gathering, data modelling and realistic display needs. The goal of his overall development has been to demonstrate the power of data visualisation approaches to efficiently gather, process, and communicate resource data. He found that the Macintosh and PC-compatible equipment widely available in forest management did not have sufficient graphics processing speed or adequate memory for the rapid graphic modelling he required. A high-performance workstation, the Silicon Graphics Indigo, was used. Programming of the visual modelling system used the C programming language and the GL graphics library (a Silicon Graphics Inc. proprietary system). The biggest problems Orland uncovered in his work were the difficulty and costs involved in creating the input databases and the ability to model and visualise very detailed changes in the micro-environment, then zoom out orders of magnitude to see those small changes in the context of regional-scale databases.

End Note

In most case studies reported in the literature the software system used is described, but it is notable that in many studies custom software is often required. It is to be hoped that the techniques developed are incorporated into software available either commercially or through user groups. However, what appears not to have been done is a quantitative evaluation of the effectiveness of one technique compared to another. Robertson and Silver (1995) call for a careful comparative assessment of different techniques, across a range of datasets to complement the experiences gained in specific case studies.

Table 3.6: Other data visualisation products

Product	Overview	Features	Data limits	Platform	Ease of use	Comments
AVS™	Application builder. Consists of 5 interactive turnkey applications. Flipbook animation.	3-D cubes, with colour adding a 4th dimension.	Limited by hardware and memory.	Runs on all major UNIX and VMS workstations.	Requires programming skills in C or FORTRAN.	From Advanced Visualisation Systems/UNIRAS Ltd. Demo licences available from: *http://www.uniras.dk/ support/faq/flexlm/demo.html*.
CrossGraphs™	Helps visualise, understand and report complex, multi-dimensional data. Can be used interactively to visualise and explore data or run in batch mode for production reporting.	Presents data in arrays of graphs. Can automatically divide data into subsets without programming. Over a dozen built-in graph components and graphical highlighting which identifies out-of-range values.	No information.	Windows 3.1, 95 and NT, Windows for Workgroups 3.11, UNIX SunOS, Sun Solaris and HP-UX.	The GUI does not require programming although it can be customised to user requirements by programming.	From Belmont Research (*http://www.belmont.com/cg*). Interfaces with ODBC-compliant sources.
DADiSP™	Interactive graphics worksheet. Results displayed in multiple windows for immediate graphic comparison.	Visually oriented software package. Instantly graphs the results of analysis.	No information.	DOS, Windows, UNIX.	Menu-driven GUI. Graphics based spreadsheet with built-in, menu-driven functions for easy use.	From Adept Scientific. Free licensed downloads available from (*http://www.adeptscience.co.uk/as /products/).*
Data Explorer™ (DX)	Advanced visualisation application development toolkit.	Similar to other application development environments. Object-oriented methodology.	Limited by hardware and memory.	UNIX platforms: IBM, Sun, HP, SGI, DG, DEC.	As with other application builders skills in programming are required to build a GUI.	From IBM Inc. More information available, in Europe from: *http://www.hursley.ibm.com/dx/*.
Data Visualiser™	Turnkey application. Keyframe and flipbook animation available.	2-D charts and overlays, 3-D displays.	Limited by hardware and memory.	Requires 25 Mb disk. Available for workstations from Silicon Graphics, DEC, SUN, IBM and HP.	Operates in GUI mode or command language mode. As a turnkey system it may be difficult for a user to add new operations.	Product of Wavefront Technologies Inc.
IRIS Explorer™	A modular visualisation environment. Support exists for functional extension and application building. Comes with 150 modules, although more are available.	Full 3-D interaction. 3-D surface and volume rendering, graphs, slicing and dicing, animation and image processing.	The only limits to data are the system and hardware configurations.	Windows NT, 3.51 and UNIX: SGI, Sun, HP, IBM, DEC.	Modules written in C, C++ or FORTRAN and connected through point-and-click interface. Requires programming expertise to build application.	Developed by SGI (*http://www.sgi.com*) and now licensed to NAG (*http://www.nag.co.uk*). User-written modules are available through international IRIS Explorer sites by anonymous ftp.
Khoros™	An application builder. Choose from 11 toolboxes, 3 needed to run program.	2-D/3-D plotting, 3-D visualisation routines, image processing, animation, interactive image display and manipulation.	Limited by hardware.	UNIX: Sun, SGI, DEC, HP and IBM X Windows. 32 Mb RAM recommended and 500 Mb free disk space.	Requires visual programming skills.	Developed by University of New Mexico (*http://www.khoros. unm.edu*). Available from Khoral Research Inc (*http://www.khoral.com*). Two versions: Khoros 2.0 Developer Release and Khoros 1.0.5. Both available through ftp, the latter is also available on CD-ROM.

Product	Overview	Features	Data limits	Platform	Ease of use	Comments
MATLAB™	Interactive system for numerical computation using matrices.	Modular package includes an image processing toolbox.	Student version limited to 32 x 32 matrices.	UNIX. Student versions for PC.	Can add extensions using C or FORTRAN. May be too advanced for most end users.	From The Mathworks Inc (*http://www.mathworks.com*). Made up of separate toolboxes.
NETMAP™	Information visualisation and data analysis tool. Allows data mining of data bases from different sources.	Interactive visualisation of graphs and charts to identify trends.	Limited by the amount of data that can be analysed at any one time. Less than a terabyte.	Variety of UNIX workstations, DOS, Power Macintosh and Windows NT.	Customisable through programming.	From Alta Analytics (*http://www.ALTAanalytics.com*).
OpenGL™	2-D, 3-D graphics and visualisation applications.	250 routines for SunSPARC, 3-D capabilities from rendering, lighting, texture mapping.	No information.	SunOS 5.3 or higher, Solaris 2.3 or higher, Open Windows 3.3.	For experienced graphics programmers, the OpenGL interface is easily understood.	From the Architecture Review Board (ARB) members: DEC, IBM Intel, Microsoft and Silicon Graphics. Information available from: *http://www.opengl.org*.
PV-Wave™	A visualisation and data analysis environment. Functions supplied but can be modified by user.	Image processing, animation, 2-D, 3-D, 4-D, surface, contour, bar plotting, colour table manipulation.	Large data sets are handled.	UNIX, OpenVMS, Windows NT and Windows 95.	Visual Exploration provides a 'point-and-click' environment allowing easy access to many of the graphics and utilities.	From Visual Numerics Inc (*http://www.boulder.vni.com*). Has been web enabled.
SPSS Diamond™	Interactive, dynamically linked windows mean changes in one window are automatically applied to others. Has statistical capability.	2-D and 3-D plotting, 4-D using colour brushing, animation, rotation, parametric snake plots, parallel co-ordinate plots, slicing.	Can start a new session with a subset of your data. Imports from a variety of file formats.	UNIX, OS/2 and Windows 3.1.	OS/2 version uses single letter commands. Windows version is more user-friendly with an easy to learn menu system.	From BMDP Statistical Software Inc (*http://www.spss.com*).
Vis5D™	Applications such as numerical solutions of the atmosphere and oceans.	Produces iso-surface plots of scalar data defined in 3-D. Designed for interactive visualisation of multi-variate, time-varying, grid data.	Does not handle unstructured data. Designed for data in grid form.	UNIX: SGI, IBM, Sun, HP, DEC.	Is extensible through programming.	Available free through ftp: *iris.ssec.wisc.edu:/pub/vis5d*
Visigraph™ using GIFIC	Excel 5.0, 7.0 add-in using the GIFIC language.	A graphing/plotting technique which allows the creation of picture displays of database information.	16,348 rows by 256 columns (Excel limitations).	Any system running Excel.	Uses 'drag and drop', 'cut, paste and link' commands. Familiar GUI to many users.	From GIFIC Corporation (*http://www.gific.com*).
WinViz™ for Lotus 1-2-3	Visual data analysis tool that complements spreadsheets and databases. Available as an add-in.	Displays multi-dimensional dataset in a single picture. Classifies data into different coloured groups.	Spreadsheet limitations. Summarises up to 8192 rows.	Windows 3.0 or later. Lotus 1-2-3 5.0 or later.	Uses the spreadsheet interface of Lotus. Point-and-click query and pull-down menus. Suitable for end-users.	Developed at the Information Technology Institute. Self-running demos can be found at *http://jsaic. iti.gov.sg/projects/vizMain.html*.

3.4. References and Bibliography

Abarbanel, R., R. M. Friedhoff, R. Langridge, J. Pearlman and J. Star. 1993. Is visualization really necessary? In Proceedings of Visualization ’93, edited by G. M. Nielson and D. Bergeron. IEEE Computer Society Press.

Banks, D. C., B. Hamann, Po-Yu Tsai, R. Moorhead and J. Barlow. 1996. Data reduction and interpolation for visualizing 3-D soil-quality data. Paper presented at IEEE Visualization ’96.

Buhyoff, G. J., W. B. White, T. C. Daniel and D. O. Hunter. 1988. Integrated computer decision support for forest impact assessment. AI Applications in Natural Resource Management, 2, 37-42.

Childers, D. G. 1997. Probability and Random Processes: Using MATLAB with Applications to Continuous and Discrete Time Systems. Irwin / McGraw Hill.

Clarke, R. T. 1994. Statistical Modelling in Hydrology. John Wiley & Sons Inc.

Cleveland, W. S. 1993. Visualizing Data. Hobart Press.

Cox, D. J. 1990. The art of scientific visualization. Academic Computing, 4, 20-56.

Daniel, T. C. 1992. Data visualization for decision support in environmental management. Landscape and Urban Planning, 21, 261-263.

Denzer, R. 1993a. Application of visualization in environmental protection. In: Focus on Scientific Visualization edited by H. Hagen, H. Muller and G, M, Nielson. Springer-Verlag, 73-84.

Denzer, R. 1993b. Graphics for environmental decision making. IEEE Computer Graphics and Applications, 3, 58-64.

Earnshaw, R. A. and N. Wiseman. 1992. An Introductory Guide To Scientific Visualization. Springer-Verlag.

Etter, D. M. 1996. Introduction to MATLAB for Engineers and Scientists. Prentice Hall.

Ford, R T., S. Running and R. Nemani. 1994. A modular system for scaleable ecological modelling. IEEE C325l Sciences & Engineering.

Forgang, A. B., B. Hamann and C. F. Cerco. 1996. Visualization of water quality data for the Chesapeake Bay. Paper presented at IEEE Visualization ’96.

Foster M. 1994. RCWP Expert, a knowledge and GIS based system for selection, evaluation, and design of water quality control practices in agricultural watersheds. Proceedings GRASS User’s Conference.

Haber, B. and D. A. McNabb. 1990. Visualization idioms: a conceptual model for scientific visualization systems. In Visualization in Scientific Computing, edited by G. M. Neilson and B. Shriver. IEEE Computer Society Press, California.

Hibbard, W. and D. Santek. 1990. Visualizing large data sets in the Earth Sciences. In Proceedings of Visualization 1993, edited by G. M. Nielson. and D. Bergeron, IEEE Computer Society Press, California.

Johannsen, A. and R. J. Moorhead. 1995. AGP: ocean model flow visualization. IEEE Computer Graphics and Applications, 17, 28-33.

King, B., P. Collins, E. Wolanski and D. Galloway. 1995. Animation techniques for visualizing coastal flow dynamics. Proceedings of the 4th International Conference on Estuarine and Coastal Modeling, edited by M. L. Spaulding and R. T. Cheng, San Diego, California.

Knapp, L. 1994. A task analysis approach to visualization of geographic data. Proceedings of NATO Workshop on Cognitive Issues in GIS.

Moran, C. J. and G. Vezina. 1993. Visualizing soil surfaces and crop residues. IEEE Computer Graphics and Applications, 13, 40-47.

Nations, S., R. Moorhead, K. Gaither, S. Aukstakalnis, R. Vickery, W. C. Couvillion Jr., D. N. Fox, P. Flynn, A. Wallcraft, P. Hogan and O. M. Smedstad. 1996. Interactive visualization of ocean circulation models. Paper presented at IEEE Visualization '96.

Orland, B. 1992. Data visualization in environmental management. Landscape and Urban Planning, 21, 237-346.

Reyment, R. A. and K. G. Joreskog. 1993. Applied Factor Analysis in the Natural Sciences. Cambridge University Press.

Rhyne, T., M. Bolstad and P. Rheingans. 1993. Visualizing environmental data at the EPA. IEEE Computer Graphics and Applications 13, 34-40.

Robertson, P. and D. Abel. 1993. Graphics and environmental decision making. IEEE Computer Graphics and Applications 13, 25-27.

Robertson, P. K. and D. Silver. 1995. Visualization case studies: completing the loop. IEEE Computer Graphics and Applications, 17, 18-19.

Rosenblum, L. J. (1994), Research issues in scientific visualization. IEEE Computer Graphics and Applications 14, 61-63.

Rossingnac, J. R. and M. Novak. 1994. Research issues in model-based visualization of complex data sets. IEEE Computer Graphics and Applications 14, 83-85.

Roth, M. and R. Guritz. 1995. Visualisation of volcanic ash cloud. IEEE Computer Graphics and Applications, 17, 34-39.

Schroder, F. 1993. Visualizing meteorological data for a lay audience. IEEE Computer Graphics and Applications, 13, 12-14.

Sheelagh, M., T. Carpendale, A. Fall, D. J. Cowperthwaite, J. Fall and F. D. Fracchia. 1996. Visual access for landscape event based temporal data. Paper presented at IEEE Visualization '96.

Stephan, E-M. 1995. Interactive visualization of environmental data. Proceedings of the ICA '95; Barcelona, Spain.

Treinish, L. 1993. Unifying Principles of data management for scientific visualization. In: Animation and Scientific Visualization Tools and Applications, edited by R. Earnshaw and D. Watson. Academic Press, 141-169.

Treinish, L. 1994. Visualizations illuminate disparate data sets in the Earth Sciences. Computers in Physics, 8, 2-9.

Treinish, L. A. 1995. Visualization of scattered meteorological data. IEEE Computer Graphics and Applications, 17, 20-26.

Tufte, E. R. 1983. The Visual Display of Quantitative Information, Graphics Press.

Tufte, E. R. 1990. Envisioning Information. Graphics Press.

Tufte, E. R. 1997. Visual Explanations: Images and Quantities, Evidence and Narrative. Graphics Press.

Waisel, L. 1996. Three-dimensional visualization of sediment chemistry in the New York Harbor. IBM

Visualization Data Explorer Communiqué, Volume 4, Number 1.

Weibel, R. 1994. Geographic Information Systems and Visualization. In: Eurographics '94 State of the Art Reports, edited by Ch. Giertsen and P.A. Fevang,: 1-31.

Wood, J. D., K. W. Brodlie and H. Wright. 1996. Visualization over the World Wide Web and its application to environmental data. Paper presented at IEEE Visualization ’96.

Yingcai Xiao, J. P. Ziebarth, C. Woodbury, E. Bayer, B. Rundell and J. van der Zijp. 1996. The challenges of visualizing and modelling environmental data. Paper presented at IEEE Visualization ’96.

4. DATA AND TREND ANALYSIS

4.1. Capabilities and Limitations

What is data and trend analysis?

By ‘data and trend analysis’ we mean statistical tools to help you decide whether the features you see in your data indicate something important or could be due to chance. From here on, we will use the word ‘statistics’ for this concept. Statistics is actually much broader than data and trend analysis, so we are fudging the concept for the sake of brevity. If we regard visualisation (see Chapter 3) as seeing the shape of data, then statistics is about analysing its mass – how many observations fall into different ranges. Statistical tools analyse variability in data – how measurements cluster together or spread out. The statistics of environmental data require special care, because variability can be caused by both instrumental inaccuracies or environmental variation. The key is to know where each is important. We will come back to this point.

Here is another way of describing statistics:

> *Statistics is the science of variability. This variability may be due to measurement errors, or the natural variability of a system (ecosystem).*

In most cases, the data are not the same as the quantity measured; they are only an imperfect reflection of it. A perfect statistic is the number of balls in a bag, which will always be the same when we count them. However, usually it is not like that; our measurement data are just a tiny and in various ways imperfect reflection of a huge and unknown reality. Statistical tools help us work with our imperfect data, to understand clearly what it tell us – and just as importantly, what it does NOT tell us – about the thing we measured. Here is yet another way of describing what statistics is:

> *Statistics provides answers to questions that are stated in the language of probability.*

Questions like:

1. What confidence does the data give that the quantity measured is less than X? (e.g. How sure are we that records on the level of toxins in drainage water are consistent with permitted levels, given instrumental inaccuracies and variations in the data?)
2. Was there a change, and if so, what confidence does the data permit? (e.g. What level of change in water quality do the data support at the 95% confidence level?)
3. What is the strength of the relationship between two quantities – their correlation? (e.g. Is there a connection between the water toxin levels and animal birth defects?)
4. On the basis of the data, what can we predict about future behaviour of the system modelled? (e.g. If toxins increased by 25% and other quantities varied randomly, what would be the predicted increase in birth defects – and how certain can we be of this prediction?)

These examples bring us to the fourth and last way of defining statistics:

> *Statistics is the mathematics of confidence. Uncertain outcomes are described by confidence levels or probabilities.*

Data and trend analysis is a set of tools for making the best possible use of measurement data, so that from the data, you can always make the best decision. Note: it does not give the correct decision, just the best one for the given data. In other words, statistics is a betting tool – it tells you the odds. The way statistics expresses the odds is through probability levels. Where a betting man would say the odds are 10:1 that Blue Streak will win the race, a statistician would say that there is a confidence level of 10/11 or 91%.

Why is it important?

Statistics is important because it helps derive the best possible information from a set of data. There are four kinds of environmental problems that statistical data and trend analysis tools can handle These correspond to the four questions mentioned earlier:

1. measuring current state of a system, expressed in some variable of interest – testing hypotheses about this state
2. detecting change of state of dynamic systems
3. quantifying relationships between variables
4. modelling and prediction.

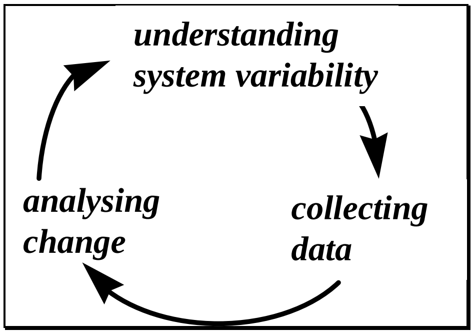
There are different statistical tools for each of these problems. Unless the appropriate tools are used (and used properly), then conclusions drawn from the data will be either too strong or too weak.

Points to be considered in experimental design include[6]:

- causal mechanisms must be checked out, not assumed away
- multiple influences on variables measured must be considered
- the likely physical response of variables over time must be considered – it is vital to take measurements at a frequency appropriate to the time scales on which the system is known or thought to be changing.

In many cases, you may not know in advance what the appropriate measurement frequencies are. Statistical techniques exist to help estimate both the sampling interval and the sample size for a particular variable. These techniques assume you have some knowledge of its parent population. This brings us to the general observation, illustrated in Figure 4.1, that data and trend analysis is not a linear process, it's a cycle.

Figure 4.1: The analysis cycle



4.1.1. Key Capabilities

In this section we survey each of the four problems, introducing the capabilities and tools that statistics offers to address the problems, and touching briefly on some of the most significant issues.

Measuring state

In theory, state means a complete set of variables that describe a system. When the system is the environment, measuring state by this definition is impossible – it is far too complex. Therefore, as

[6] A related area in which statistics is central is experimental design, or 'taking the right measurements'. The cost of gathering information is high and proper use of experimental design tools can minimise this cost. This aspect is covered only briefly in this manual, but readers are referred to the wealth of literature on the topic, for example Kuehl (1994).

many variables as possible are measured. Ideally, state refers to a single instant in time, but in practice for ecosystems, it often refers to an extended period: a day, year, or even longer.

Measurements of state can be summarised with a statistic indicating the level of the parameter of interest (e.g. mean, median, mode) and a statistic to indicate the range of variation of the data about this level (e.g. standard deviation, minimum, maximum, distribution quartiles). Information on the variation allows us to derive confidence intervals for the state, and is crucial to other forms of testing hypotheses about the state.

For measuring state, the key capabilities are:

- finding the parameters of the data distribution (mean, median, standard deviation, quartiles, minimum, maximum)
- testing residuals to see if the measurements are independent and satisfy other assumptions necessary for confidence interval calculation
- establishing confidence intervals on the parameters.

The tools that address these needs are:

- the basic functions offered by all statistical packages to find the parameters[7]
- residuals plots to check the independence of the measurements
- distribution tables (or alternatively the bootstrap see below) to establish confidence intervals.

Detecting change

When we measure state, we implicitly assume that there is a single state – that the system itself is stable. This introduces some hazards. How can we find out what the data are telling us about how stable the system is? Is there a single state, or are there more than one? More generally, in a continually changing environment, how is it possible to establish that a change of some magnitude has occurred? And if we think that it has, how confident can we be? What if there is not a set of distinct states, but rather gradual change?

For detecting change, the key capabilities are:

1. discriminating among states, to identify possible changes
2. hypothesis testing, to establish confidence that the states are distinct.

The tools that deliver these capabilities are (1) the box plot and data distribution to identify candidate states and (2) the t-test and related tests, to establish confidence levels in particular values of the state.

Quantifying relationships

A relationship among variables is indicated when a scatter plot of the variables shows a tendency for changes in one variable to be matched by changes in another. The left panel in Figure 4.2 shows a relationship between a hypothetical measurement variable (OP) and time. On the right is one between two measurement variables (OP and N). The data are plotted as points, and a single line is drawn to show an interpretation of the data. These lines show simple linear relationships between variables.

In Figure 4.3, a more sophisticated set of lines is used which might more closely describe the same data. In both panels, we see from the data that the nature of the relationship changes at a certain point: on the time series plot the level of OP appears constant, then rises, then levels off again. The OP-N relationship appears to be linear (OP = const + slope * N) for low levels of N, but then the slope seems to increase dramatically.

[7] We advise against the use of spreadsheet tools for statistical calculations; while they may suffice for brief analysis tasks, the ones we have considered frustrate a user by providing only part of what is needed.

Figure 4.2: Scatter plots illustrating simple linear relationships

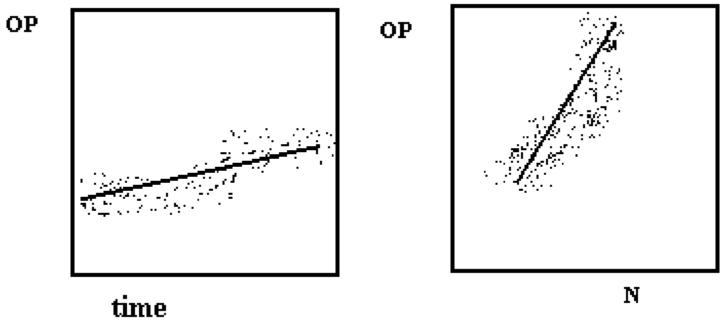
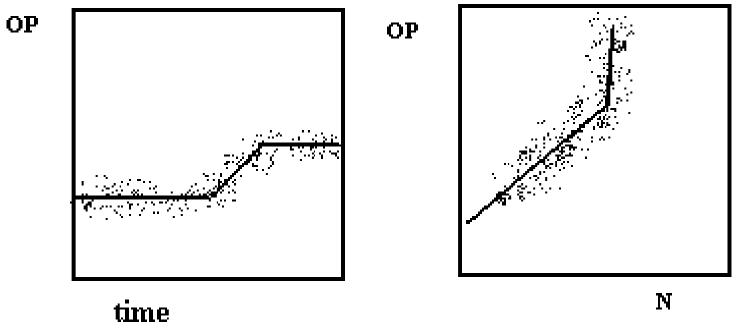
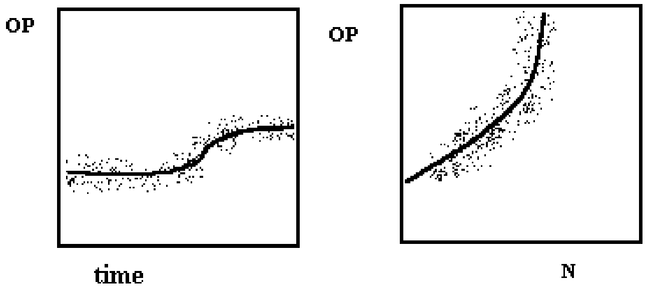


Figure 4.3: Scatter plots illustrating changing relationships



These illustrations suggest a number of important considerations in quantifying relationships with statistical tools. First, finding an equation to describe the relationship. The simplest possible relationship consistent with the data is sought. Knowing when to stop adding parameters to a fitting function is the issue. There are test statistics to help with this. Second, distinguishing among possibilities. The lines drawn on the data in the above examples are just one possibility. Some alternative relationships that could be used to describe the same data are shown in Figure 4.4. These are spline curves that represent a further increase in the level of sophistication from those shown in Figure 4.2 and Figure 4.3. Statistical tools can help us decide among alternative relationships – or alert us to the fact that the data are equally well represented by different relationships. A third issue is knowing how well an equation describes a relationship. An equation can be a ‘best fit’ to data, but still not be a good description of it. For example, if the underlying reality in the right panel of Figure 4.4 is that there are two parallel curves. This could occur for example when one of several instruments being used to make the measurements is mis-calibrated leading to a set of results different from the remaining instruments by a more or less constant offset. Statistics could tell us that the fit to the data was poor for the single curve leading us to try parallel curves. It might not be at all obvious that this is the case from viewing the original data or using simple linear relationships. But a note of caution: while statistics can help us ‘dig out’ measurement anomalies such as those in this example they are not full-proof means of doing so. The physical reality of the measurement scenario must also always be considered while analysing data.

Figure 4.4: Scatter plots illustrating spline curves



This is a good point to introduce another more general caution in the use of these statistics.

> *Real physical mechanisms must exist (proved or hypothesised) linking variables within the environmental system being investigated to provide an explanation for statistical relationships. The links may be direct or indirect (e.g. two variables are related but the causal mechanism is provided by a third variable).*

This does not mean that the analyst should never investigate relationships between variables for which no direct (or indirect) physical mechanism is known; but that underlying mechanisms must be considered as part of the analyses. Without such checks, false conclusions can be reached.

In the above example, the relationship between OP and N could be assumed to be dependent or causal (e.g. an increase in the N concentration causes an increase in OP concentration), but this is almost certainly not true. In reality, other factors probably control both OP and N concentration.

To summarise, quantifying relationships requires three statistical capabilities: determining if there is a significant relationship, fitting model parameters, and assessing the quality of the model. The tools that deliver these statistical capabilities are (1) correlation coefficients, (2) linear and non-parametric regression and (3) goodness-of-fit statistics such as chi-squared. We will not pursue these topics further here; for details consult any of the references listed in Section 3.4.

Modelling and prediction

Once there is an understanding of the nature of the variability in the data, the next stage is using this understanding to predict unobserved states. This is either in the future (in the case of time-series data), at a different location (for spatial data), or when one or more of the measurement variables changes. Given a model – a fit to data such as described in the last section – it is easy to see what the model predicts for a future time or a different set of inputs. What statistics adds at this next level is the ability to assign confidence levels to predictions. To make statements like these:

- there is a 95% probability that tomorrow's temperature will lie between 20 and 22 degrees
- the probability of a nitrate concentration less than 20 ppm being found in the drainage channel two days after an application of 100 kg fertiliser to the field is 10%.

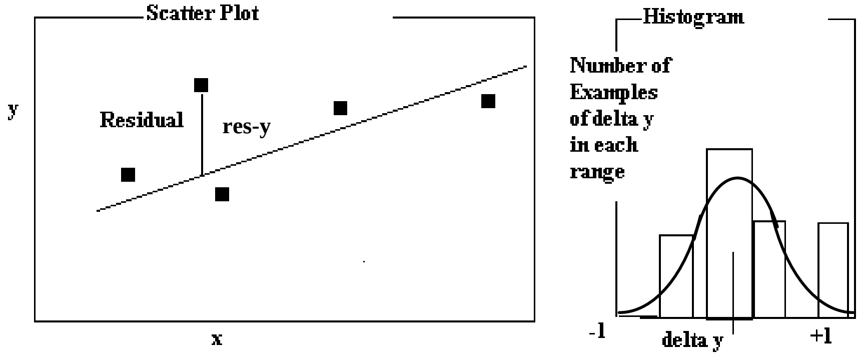
The key capability for modelling and prediction is estimating uncertainties associated with predictions. The tools that deliver this capability are (1) error analysis and propagation and (2) bootstrap sampling for specific models, e.g. time series.

Analysis of residuals

There is one other core aspect of these kinds of CIPTs that distinguish them from the others in this manual: they attempt to extract ALL the available information from the data. In particular, great importance is attached to analysing the distribution of the residuals. To explain, consider

Figure 4.5, for each point on the plot, the difference (res-y) between the y co-ordinate of the point and the y co-ordinate of the line at the same x is the residual for that point. In simple terms, if the line is a good fit to the data and the data are normally distributed, then a histogram of the residuals will be a normal distribution with zero mean (see Figure 4.4). Of course, how much like a normal distribution the histogram looks depends on the amount of data; to illustrate this point, Figure 4.5 shows just a few data points.

Figure 4.5: The histogram of residuals



The main issue here is that residuals should be examined. The statistics of the residual distribution (mean, median, shape, etc.) can indicate that the fitted line is appropriate or not. Tests can be applied to the distribution of residuals to verify that the assumptions that have been made about the data are true, or more precisely, to estimate the probability that they are false.

4.1.2. Limitations and Likely Evolution – the Authors' View

What is wrong?

Even with the best of statistical CIPT products, it is easy to make mistakes with statistics. Although appearing simple it is a rather sophisticated topic, and unfortunately it takes a relatively long time to become proficient in selecting the right tool for the data and the task. The products could be improved in a number of ways:

- the learning curve is too steep and long – the tools should be embedded in a framework that guides the user in their use
- data preparation aids are limited – tools to import data from all kinds of sources are needed
- the tools rarely handle datasets satisfactorily – they should understand the limits of the data and guide the user as possibilities are explored
- the best of visualisation should be combined with active statistical tools that allow users to make hypotheses and see their consequences in an integrated fashion
- statistical capabilities ought to be embedded in a larger problem-solving framework that incorporates other CIPTs.

What is happening and where is it heading?

The best currently available tools already take steps in the directions indicated above. In particular there is considerable incorporation of data visualisation, for example in S-Plus (see Section 3.2.2). In the short term, there will be an increasing availability and ease-of-use of computer intensive statistical methods. It is likely that there will be a convergence of visualisation and data analytic tools, to allow early exploration followed by fitting and testing of data features. Tools specific to various application areas are already emerging, often built from some base system (such as the spatial statistics add-on for S-Plus) and this is likely to become more widespread.

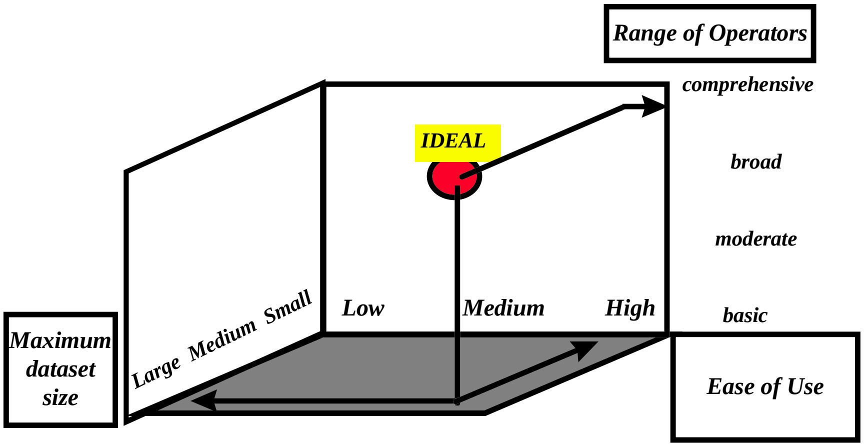
4.2. Representative Packages

Figure 4.6 shows three of the factors that are most important in selecting an appropriate statistical package for an application. An 'ideal' tool would handle as large a dataset as we would like and allow us to usefully employ a comprehensive range of operators with high ease of use, i.e. minimum learning time and high productivity. Some other factors that may be important are cost, availability, speed and the quality of the documentation. We will discuss three statistical packages, showing where their capabilities place them on the axes of the diagram and mentioning other factors as they arise.

The packages are:

- Minitab – a simple statistical package
- S-Plus – an object oriented package
- SAS System – an advanced suite of analysis tools.

Figure 4.6: Statistical package selection factors



This is followed in Section 4.2.4 with a listing of other systems currently on the market. As you will see, in the currently available packages, ease of use means limited range of operators, and vice versa. The packages are generally not suitable for handling datasets that approach 1 million or more records.

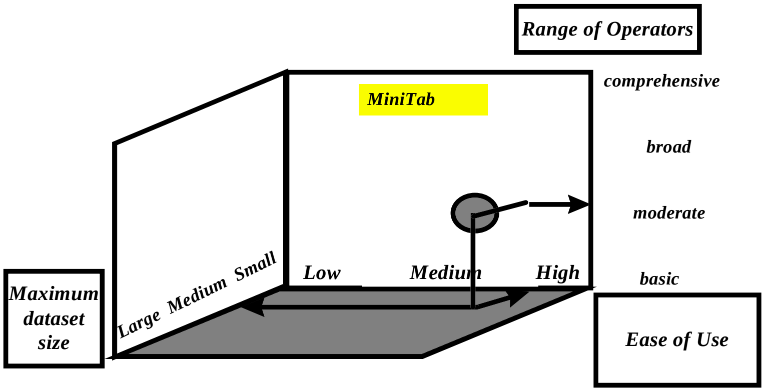
4.2.1. Minitab™ – a simple statistical package

Designed in the 1970s as a teaching tool for students with no knowledge of computing, Minitab was slow to modernise, offering very basic statistical techniques in a user-friendly way. Gradually, Minitab began to offer a broad range of functions, whilst still being a simple tool to use. Statistical tests not already offered by the package can be programmed using the BASIC-like macro language making Minitab a comprehensive tool for statistical analysis. Minitab is available on all popular platforms and is available in English, French and Spanish. Its power is illustrated in Figure 4.7.

Range of operators

Minitab is a good tool for exploring data without having to know about computing. It offers over 200 statistical analysis capabilities. Minitab recognises three data structures: columns, constants and matrices. It has a good range of descriptive statistics including regression analysis, correlation, time-series analysis, analysis of variance, simple hypothesis tests of both parametric and non-parametric types, multi-variate analysis including principal component analysis, factor analysis, cluster analysis and options for designed experiments. Minitab macros allow the analysis to be customised using IF-THEN-ELSE statements.

Figure 4.7: Minitab™ selection factor ratings



Maximum dataset size
Minitab is suitable for dataset sizes up to a maximum of a few thousand records.

Ease of use
Due to its conception as a teaching tool, Minitab is very easy to use. Data can be entered in a spreadsheet-style and results and summaries viewed in easy-to-understand plots and charts. The graphical user interface (GUI) makes the interactive environment even more user-friendly with pull-down menus and drag and drop options. There is a command-line option and macro programming available for more advanced users. The on-line help has been expanded to accommodate these new capabilities. The GUI allows the user to select commands from a pull-down menu. Data can be imported directly from Excel, Lotus, Symphony, Quatro Pro, dBase and text files. Minitab also supports Open Database Connectivity (ODBC); a capability that allows you to get data from other ODBC-compliant packages such as Access, Oracle, Sybase or Informix. Commands to read and write data from files are available, as well as a spreadsheet-style layout and full screen editor. Traditional summary statistics and graphics are available to visualise data. Graphical forms include different types of scatter diagrams, scatterplot brushing and tools to customise and edit the graphs to presentation standard. Graphs are automatically produced for quick analysis. A new feature of the Macintosh version is the 3-D surface plot with hidden surface removal, lighting, positioning and viewing options.

Learning time
Minitab has a well-developed help system that can be referred to at any stage. It is possible to learn to use Minitab from the help system alone and it is one of the few statistical packages that can be used without the aid of manuals. The friendly user-interface and the simplicity of the package make it easy to learn for those with no computing knowledge. One can become familiar with Minitab in a matter of hours. In addition to documented help on-line, manuals, reference guides and a Getting Started tutorial are available. As the software is widely used by academic institutions, the WWW sites of many of these institutions have on-line reference guides and tutorials for Minitab users. Examples of these are:

http://www.leed.ac.uk/ucs/docs/beg20/beg20.html
http://www.cardiff.ac.uk/uwcc/comp/docs/pc/

Cost and availability
Minitab is available directly from its producers; Minitab Inc. (*http://www.minitab.com*) and through regional offices. Registered users can obtain telephone support from technical specialists during office hours or on-line support from Minitab Inc through their website at /. Information on training is also available. The system requirements are: Windows 3.1 or later, Windows 95, Windows NT 3.51 or later, DOS 5.0 or later. Suitable platforms include 386 processor or above with 20 Mb hard disk space for full installation; and 68020, 68030 or 68040 Macintosh and Power Macintosh with 8 Mb RAM and 21 Mb hard disk for full installation. Also available as Minitab

Release 8 for DOS, Minitab Release 9 for VAX/VMS and ALPHA/VMS and Minitab for UNIX. Table 4.1 shows the cost factor rankings for Minitab, using the categories defined in Table 2.1.

Table 4.1: Cost factor rankings for Minitab™

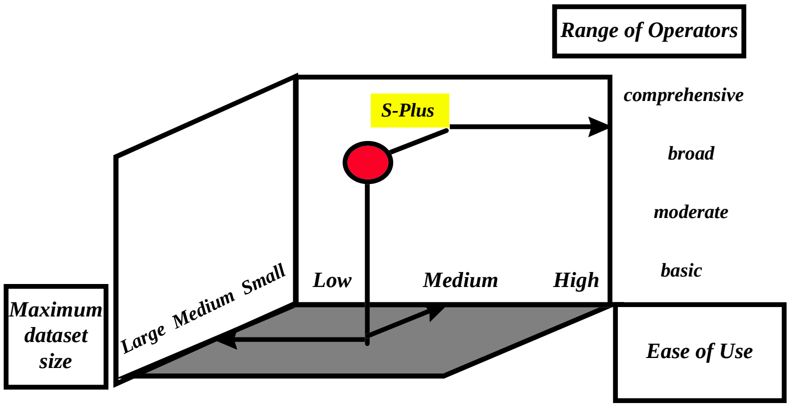
learning	installation	application	price
B a few hours	A less than 10 minutes	A tens of minutes	A

4.2.2. S-Plus™ – an object-oriented package

Many leading academic organisations are using 'S'; an object-oriented programming language from AT&T, which is designed specifically for data analysis. S-PLUS was developed by the Data Analysis Products Division of MathSoft Inc. and is an object-oriented exploration and data analysis package that is written in (and expects the user to know) S. With over 1,650 functions, it can accomplish most tasks and gives the user complete control. S-Plus gives immediate feedback after every step allowing the output of one operation to be used as the input to another. The ratings for S-Plus are shown in Figure 4.8 and discussed below.

Range of operators
This package allows exploration of data, interactive analysis through a wide range of functions, visualisation and modelling. It covers parametric and modern non-parametric methods. It has a built-in object-oriented language and can interface with C and FORTRAN programs. The S language on which S-Plus is based is a flexible, vector-based mathematical language that supports matrices and higher dimensional arrays. This makes it ideal for developing mathematical computing applications, simulations, or new algorithms.

Figure 4.8: S-Plus™ selection factor ratings



The functions for performing and managing data analysis include basic statistics such as the chi-square test, as well as probability distributions, multi-variate statistics of hierarchical clustering, tree classifiers, principal components, regression, ANOVA, survival analysis, time series, signal analysis, Fourier transforms, classical and robust auto-regression models and auto correlation. If the available functions do not match the user's needs exactly, the S-Plus tools can be modified using object-oriented programming techniques. Extra modules are available through specialised academic networks, such as that for spatial statistics.

Maximum data set size
S-Plus is suitable for medium-sized datasets, i.e. tens of thousands of items, not millions.

Ease of use
Although extremely powerful, this package is probably not recommended for novice or occasional users. S-Plus primarily uses a command-line interface with UNIX-like commands. There is an extensive on-line help system with good indexing, but it assumes that the reader is experienced in statistics. Once familiar with the interface, ease and speed of use can be very good; although to get used to the command-line can take some time. Graphics in S-Plus are interactive and can be displayed as 1-D, 2-D or 3-D. Features include multiple graphics windows, point identification using the mouse, interactive colour maps, 3-D data spinning, scatterplot matrix brushing, 2-D contour plotting, overlaying and general mapping functions. S-Plus features customisable menus, dialogue boxes and controls to generate S-Plus commands. The Windows version has an interactive point-and-click help system, extensive documentation, command line recall and editing, telephone and e-mail helpline. Advanced functions developed by scientists and mathematicians using S-Plus are available by e-mail or through the Internet.

Learning time
The need to use a programming language adds to the package learning time. The S-Plus dialogue boxes are rather less friendly than the point-and-click packages.

Cost and availability
S-Plus is available from StatSci Europe the leading distributors throughout Europe (*http://www.statsci.co.uk*). The system requirements are Windows 3.1 or higher, MS-DOS version 5.0 or later with 8 Mb RAM, 14 Mb free disk space for minimum installation or 25 Mb free disk space for complete installation; SUN Sparc 4.1.x and SUN Solaris, IBM RS/6000, Hewlett-Packard 9000 Series, Silicon Graphics, or DEC Alpha. X-Window system (Motif) with 16 Mb RAM and 65 Mb free disk space. Table 4.2 shows the cost factor rankings for S-Plus, using the categories defined in Table 2.1.

Table 4.2: Cost rankings for S-Plus™

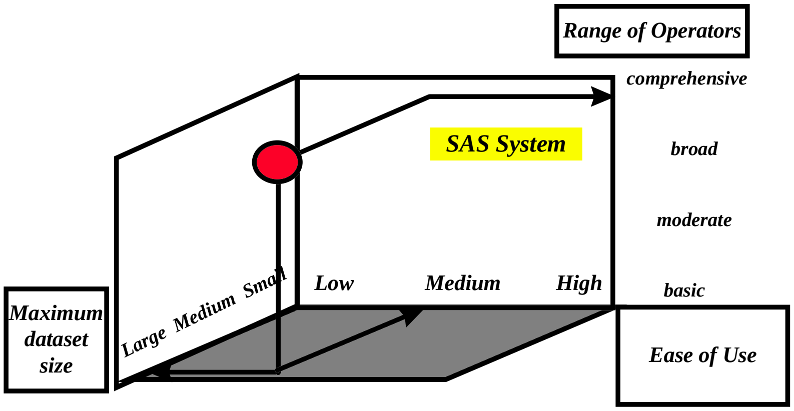
learning	installation	application	price
D weeks or months	B about an hour of a specialist	C one to several days	D plus annual licence

4.2.3. SAS® System: an Advanced Suite of Analysis Tools

SAS® System, from SAS Institute, is a large collection of software modules that encompass data access, storage and management, analysis and visualisation. SAS System is well suited to those needing a complete working environment, from planning their work through to producing final reports. It can be used in interactive, non-interactive and batch modes. The overall range of facilities available is impressive, but in some areas the particular task can be accomplished better by specialised packages. This must be weighed against the benefit of an integrated environment. Our ratings for SAS System are shown in Figure 4.9 and discussed below.

Range of operators
SAS System has a full programming and macro language, together with in-built Structured Query Language (SQL). It offers a broad range of statistical and mathematical functions and statistical procedures such as econometric procedures, time-series analysis and experimental design. Its high quality, interactive graphics enable data to be thoroughly explored and well presented. The SAS/STAT module provides procedures for linear and non-linear regression, analysis of variance, multi-variate analysis, survival analysis, response surface regression, logistic regression and covariance structure analysis. It also features a very full Generalised Linear Interactive Modelling (GLIM) implementation. Other modules within the SAS System include a spreadsheet, an interactive statistical graphics module, interactive guidance-driven analysis, and querying tools.

Figure 4.9: SAS™ System selection factor ratings



Maximum dataset size and data type
The SAS/ACCESS module provides an interface to various file formats including PC file formats, OS/2 Database Manager, Sybase, Oracle and others. SAS/ACCESS software enables most relational databases to be used directly, without the need for creating external copies. Import and export wizards make data import and export a simple matter. Thus the maximum dataset size is limited by the database. SAS System can read data in numeric, alphanumeric, binary, dollar, date and time formats.

Ease of use
Reflecting its substantial history, most of the SAS components do not come packaged with a GUI, but rather have GUI modules that you buy separately. An example is SAS/AF, which offers customised help facilities, menu and dialogue windows. SAS/INSIGHT software (an exploratory data analysis and interactive statistics tool) has a full GUI. Other graphical front ends are available for specific functions. Advanced statistical, engineering and mathematical analysis programs can be written using the SAS Interactive Matrix Language (IML), which resembles PL/1 or C. For users with a need to use matrix operations, SAS/IML provides the ability to program using matrices as objects. The SAS/ASSIST module provides a ‘point and click’ front end that requires no knowledge of SAS code, and this generates fully annotated code that may be re-used. With Release 6.12 the SAS desktop enables the user to customise the exact functionality they need. Other tools are provided for the user to develop custom user-friendly environments. SAS System offers graphics modules for data visualisation and presentation. Data can be explored through interactive histograms, box plots, scatter plots, 3-D rotating plots and geographical maps.

Learning time
SAS has a steep learning curve, especially as it is programmable. To help ease this, SAS/TUTOR software is a range of computer-based introductions to using SAS for first time users, and there is a range of learning aids. The full SAS documentation suite occupies a large bookshelf, but most users will select a small fraction of this to meet their immediate needs. On-line help and telephone support are available. The SAS Institute supports a Books by Users programme and there is a large library of such titles. There are several annual SAS conferences and international and European user groups.

Cost and availability
SAS software is only available directly from one of the SAS Institutes *(http://www.sas.com)*. It is subject to an annual licence fee, which include hot-line technical support, documentation and automatic upgrades. SAS System runs under DOS, Windows (3.x, 95 or NT), OS/2 and on all major UNIX platforms. 16 Mb RAM is recommended for Windows, with 120 Mb of disk space being adequate for most needs. Table 4.3 shows the cost factor rankings for SAS System, using the categories defined in Table 2.1.

Table 4.3: Cost factor ranking for SAS® System

learning	installation	application	price
D several weeks or months	B about an hour of a specialist	C one to several days	F plus annual licence

4.2.4. Other Products

Table 4.4 provides summary details of some of the other data and trend analysis tools that are on the market today.

4.3. Environmental Applications

There are numerous sources of information on the use of statistical tools and it is recommended that you locate ones in your particular discipline area – a few specialist and general text books are listed under the application examples later on. Most vendors provide excellent information about their tools on their web-sites and some list books and articles to support their packages, i.e. Anthony (1996), Earickson and Harlin (1994), Kuehl (1994), Ross (1996) and Shaw and Wheeler (1994) all describe using the SPSS package. In addition some web-sites have worked examples, such as S-Plus *(http://www.statsci.co.uk/s-plus/stsplus.htm)*, which includes an interesting analysis with S+SpatialStats of a coal ash data set. In contrast more scientifically oriented publications often only describe the statistical techniques themselves, e.g. principal components, regression analysis, etc., rather than the computer tool. However, such literature, which includes the periodicals Applied Statistics, Applied Stochastic Models and Data Analysis, Biometrica, Communications in Statistics and the Journal of Agricultural, Biological and Environmental Statistics provide a rich discourse on the latest work.

To illustrate the use of data and trend analysis tools for environmental applications we have selected some examples from two of these sources, namely the vendor's web-sites, where the benefits of a particular package are highlighted and from the scientific literature, where the specific tool used was mentioned. The style of these two approaches are markedly different, but it is hoped they provide some guidance on the capabilities of the various packages. The examples included here cover:

- urban micro-climate analysis
- biological diversity
- agriculture
- environmental quality assessment.

Urban micro-climate analysis

Students at Gaithersburg High School in Maryland used Minitab to study the urban heat island effect in the Washington D.C. area and in particular to ascertain if the early-morning temperatures varied across the urban area *(http://www.minitab.com/ap-kraye.htm)*. For eight months, 20 students spent about two hours each day charting the daily temperatures in the metropolitan area, collecting weather maps which showed pressure systems, fronts, and storms and using a tethered weather balloon measuring low-altitude changes in temperature. With all of this information to be analysed, the project co-ordinator, had to find a way to efficiently collect and organise the data. Minitab was recommended because it is easy to work with, especially for researchers with very little background in statistics. The team used Microsoft Excel to enter the data and then imported it into Minitab, which enabled them to analyse their data and come to conclusions as to why the climates differed.

Biological diversity

Like many island nations, Madagascar is facing tough decisions about how to conserve its unique biological diversity, while also accommodating the needs of its growing human population. The Peregrine Fund is helping conservation authorities in Madagascar make conservation decisions by supplying quality information on which to base those decisions *(http://www.clearsoft.com/newsltrs/r_d/rdv2n1/mad.htm)*. The research team have been using SYSTAT to statistically and graphically analyse a wide variety of data collected in the field, from sociological and economical issues faced by local people, to population analysis of the endangered Madagascar Fish-Eagle, to habitat models in the rain forest. Daily weather data were entered and SYSTAT used to analyse mean, minimum and maximum values, standard deviation, etc., on a monthly basis. Overlay charts made it easy to see the driest, wettest, hottest or coolest months of the year. Such information, in relation to food source availability, was useful in determining the size and placement of a new National Park to ensure the survival of viable populations of endangered species within small and fragmented patches of forest. One factor in selecting SYSTAT for this study was its capacity to run on desktop PCs – an important consideration for field operations.

Agriculture

The sheer volume of livestock production in some areas of the world is outstripping traditional methods of waste treatment, namely lagoon retention and spreading on the fields. To address this problem Cressie and Majure (1997) used Arc/Info and S-Plus for all the statistical analysis to develop a spatio-temporal statistical model of log nitrate concentration from daily data collected over a 15-month period. The model allowed the prediction of contamination concentration in space and time with a known confidence. This relatively novel approach to use a GIS and statistical modelling, included weighted regression, variogram estimation, variogram model fitting and kriging.

For an analysis to predict characteristics of the new crop of apples Ming-Hui Chen and Deely (1996) utilised a Bayesian methodology with the Gibbs sampler. They used a packages known as BUGS (freely available by anonymous ftp from *ftp-mrc-bsu.cam.ac.uk* in directory pub/methodology/bugs) to perform a constrained linear multiple regression, which was found to be superior when compared with ordinary and inequality constrained least squares estimations. Most models used to predict the presence/absence of a disease in agricultural crops are based on continuous response variables. In contrast Gumpertz et al (1997) looked at binary response variables (an autologistic model). They used pseudo-likelihood estimation with parametric bootstrap standard errors and existing statistical software (in their case SAS). They concluded that methods of estimation for logistic regression models will rapidly become more available, with commercial software appearing within the next several years.

Environmental quality assessment

The utilisation of indicators is an important tool in the assessment of environmental quality. Dominici et al (1997) used an approach combining Baysesian hierarchical modelling and a data augmentation model to investigate the level of chlorophyll-a – one of the most important indicators of lake water quality. For further information also see *http://www.isds.duke.edu/~gp*. Whereas, Qi Jian and Succup (1996) used a Monte-Carlo approach to investigate the applicability of the log-log and log-additive models in predicting children's blood-lead levels. The relationships between blood lead and the amount of lead on hands and the concentrations of lead in soil, paint and dust were the major research concerns. The study recommended that the log-log model be chosen for application to epidemiological data as although a plethora of environmental phenomenon may be linear in their variables, lead exposure does not appear to belong to this category.

A major application area where trend analysis is particularly critical is in the monitoring of radiation levels. To finish this applications section we describe two major operational programmes, one in the USA and the other in Germany.

Virgil Parola, senior equipment technician for the state of Illinois uses SYSTAT for Windows in a nuclear radiation detection programme *(http://www.clearsoft.com/software/science/solutions/nuclear.html)*. Before using SYSTAT, he manually plotted water radiation and contamination data and did minimal data reduction. He was unable to verify distributions or analyse large groups of data, however, because he was limited to basic mathematical models. “I noticed then,” Parola explained, “there was something missing. It was a game at times to get a feeling as to what would happen. I would have to take my best guess. One needs to understand the randomness of radiation and nature. The whole concept lends itself to statistical evaluation. Once I started testing, I knew I needed a good statistical package”. He collects data every quarter from approximately 450 radiation detectors placed throughout the state and works with massive amounts of data. When he first started to use the package he did not think he would ever use all its functionality, now he says “I really enjoy the versatility.” He performs data interpretations to see if the data fit the appropriate model. “Distributions in nature tend to be log-normal. If the data don’t fit the model, then I know I should review my processes or the physical location of the detector.” From his research, Parola has found several factors that affect the data: for example, soil content. In cities, there are excess amounts of thorium from waste materials used as fill and in rural areas, fertiliser used for farming increases the amount of potassium in the soil. Both of these additives increase the amount of radioactivity, and therefore, the radiation levels. To account for these variables, Parola uses SYSTAT as a refining tool through repetitive data analysis in order to best approximate the predicted distribution and identify any outliers.

In December 1986, following the Chernobyl disaster, the German government passed the Precautionary Radiological Act to ensure radiation levels are measured regularly and in a standardised way throughout the country; so that in the event of any abnormal increase in radiation, relevant information is made available to the appropriate authorities (SAS Institute, 1993). Sixty regional measurement agencies monitor radiation levels in substances such as sewage or soil and foodstuffs for animal and human consumption. This data is sent to the national monitoring bodies for checking before being forwarded to Bundesamt für Strahlenschutz for analysis in an Integrated Measurement and Information System (IMIS). IMIS, which outputs about 50 different pre-defined tables, graphs and maps is based on the SAS System for data analysis, visualisation and presentation. Reiner Buzin who was responsible for implementing the SAS System within the IMIS project explained they “wanted a system which was very user friendly, which minimises the time [the scientists] spend getting at the data, and maximises the time they spend working with it”. In brief, the SAS System was chosen for its data visualisation and presentation features and its mapping facilities. The ability for scientists to access the code behind the applications and import external data into IMIS were also important.

Table 4.4: Other data and trend analysis products

Product	Overview	Features	Data limits	Platform	Ease of use	Comments
DADiSP™	For visualisation and data analysis. Add-on modules available for statistics, neural nets and advanced digital signal processing.	Has 500 functions, works with data series, matrices, images, waveforms, signals and includes SPL (Series Processing Language).	Commercial version handles vast amounts of data. A free student version accepts data series of 8192 points.		Intuitive, spreadsheet-like, icon and menu driven environment for displaying and analysing data. Programs built by linking analysis cells without programming.	From DSP Development Corporation (http://www.dadisp.com).
DataDesk™	Based on exploratory data analysis, emphasising visual, interactive tools for finding patterns, trends and outliers. Version 5.0 includes an object-based 'Action' programming language and custom automated templates.	Summary statistics, non-parametric tests, cluster analysis, multiple regression principal components, a general linear model supporting ANOVA, MANOVA, repeated measures and other designs. Graphics include 3-D rotating plots.	Features an unlimited number of cases and variables. The developers claim DataDesk to be the fastest data analysis program on the Macintosh.	Any Macintosh with at least 1 Mb of RAM. It is Power Macintosh native. Comes with over 800 pages of documentation, a help system and free technical support.	Drag and drop interface makes changing variables easy and lessens the learning curve. Only a few hours needed to learn the interface. Programming ability increases functionality.	From Data Description Inc (http://www.datadesk.com).
Gauss™	Matrix programming language used interactively or in edit mode. Add on modules: time series, maximum likelihood, linear regression.	High resolution 2-D and 3-D publication quality graphics. Creates plots, histograms, box graphs. Unlimited combinations.	Matrix size unlimited. DOS limited to 8192 elements.	Windows 95, NT and DOS. OS/2, UNIX. Windows needs 12 Mb RAM and 6 Mb for installation.	Requires programming knowledge with a steep learning curve.	From Aptech Systems (http://www.aptech.com). It has almost the same level of functionality as SPSS and SAS.
Genstat™	GENeral STATistics package. Interactive with a menu system and command language. High resolution graphics include contour plots, dendogram and 3-D surface plots and histograms.	ANOVA, experimental design, linear, non-linear and generalised linear modelling, multi-variate analysis, survival analysis, geostatistics and time series analysis.	No information available.	Windows, Digital VAX/VMS and UNIX workstations. Product support is automatic except for users running Genstat 5 on PCs, for which full support is optional and incurs an extra charge.	Genstat for Windows has user friendly menu-driven GUI. Programming is required to customise package. Is a single package, not a collection of modules.	From Numerical Algorithms Group Inc. (http://www.nag.com) Offer variety of product pricing and licensing options. Genstat newsletter and discussion list.
Instat™	Performs basic statistical calculations. One-way ANOVA, non-parametric tests, contingency tables, linear regression and correlation.	Bar graphs, scatter plots, but it does not facilitate more advanced graphic presentations.	Analyses very small data sets (26 columns x 500 rows).	DOS 3.0 or higher, which can run under Windows, Macintosh system 6.0 or later.	Very simple to use. Explanations of statistical tests and their results are provided.	From Graphpad Software (http://www.graphpad.com). For basic analysis. Graphpad Prism includes InStats' statistical features plus non-linear regression and Kaplan-Meier survival analysis.
Maple V™	A powerful mathematical problem-solving and visualisation system. Complete procedural programming language with conversion to FORTRAN or C code.	Calculus, equation solving, elementary and special functions, linear algebra, and ANOVA. Graphics includes 2-D and 3-D plotting and interactive 2-D and 3-D animation.	No information available.	Windows 3.1, 95, NT and OS/2 with 8 Mb RAM and 18-30 Mb hard disk. Macintosh and PowerMacintosh version 7.0 with 4 Mb RAM and 14-21 Mb hard disk. Earlier releases are available for a variety of workstations.	A high-end product. Toolbar buttons lack annotation describing their function making the interface difficult to learn.	From Waterloo Maple (http://www.maplesoft.com). Maple User Group can be subscribed to and a share library accessed by anonymous ftp or electronic mail. Software demonstrations available at web site.

Product	Overview	Features	Data limits	Platform	Ease of use	Comments
Mathe-matica™	Handles numerical, symbolic, and graphical computations and has a built-in programming language.	Used for visualisation, modelling and data analysis. Capabilities include symbolic matrix functions, 3-D visualisation, animated graphics, pattern matching and differential equations.	Limited by system memory, i.e. suitable for moderately large datasets.	Windows 3.1, Windows NT, Macintosh, Power Macintosh, DOS, IBM OS/2, NEXTSTEP and the X Window System.	Programming necessary. Application packages available to call Mathematica directly from Excel. Notebook user interface available for some platforms.	From Wolfram Research (http://www.wri.com) and distributed by Rapid Data Ltd. (e-mail: info@radata.demon.co.uk).
MATLAB™	Interactive system for numerical computation using matrices. A number of toolboxes are available, the Statistics Toolbox provides a collection of functions for analysing, characterising and developing algorithms.	Principal component analysis, response surface modelling, multiple regression, multi-variate statistics, parameter estimation, non-linear data fitting, statistical process control, experiment design and statistical plotting.	Limited by system resources, i.e. suitable for large datasets.	Available for leading desktop engineering platforms on which MATLAB is supported, including PCs, Macintosh, UNIX and VMS workstations.	Can add extensions using C or FORTRAN. The Statistics Toolbox includes demos of experiment design, response surface modelling and non-linear curve fitting and prediction. The toolbox offers a point-and-click environment for easy use.	Developed by Mathworks Inc (http://www.mathworks.com) with distributors in most European countries. Designed specifically for non-statisticians
Metrica™	Application development tool for data management, analysis and visualisation of 3-D and 4-D surface plots.	Hundreds of mathematical, statistical, and signal analysis routines and user-defined functions. Applications can be customised using Metrica's Technical scripting Language, an interpreted language.	Large data sets can be handled as analysis routines are executed on the server –only the results are passed back to the client.	Runs on leading UNIX workstations. Has C, FORTRAN, Pascal and BASIC program interfaces for direct access to server functions.	Has a point-and-click interface. Database queries can constructed by selecting on-screen columns. Knowledge of programming enhances Metrica's abilities.	From Metrica (http://www.metrica.com).
SigmaPlot™	Technical data analysis and graphing package. Can open an Excel spreadsheet directly within SigmaPlot. Uses the functionality of Excel and directly creates SigmaPlot graphs. The mathematical transform language allows automation of complex routines.	Transforms including one-way ANOVA and fast Fourier, non-linear regression with up to 10 independent variables and 25 parameters, linear regression, Boolean or weighted functions. Graphics include 2-D plots and time series and 3-D line, contour and rotation.	Over one billion data point worksheet (16,384 columns and 65,436 rows).	Windows 95 or NT with minimum 8 Mb RAM (16 Mb recommended) and 13-25 Mb free disk space. Windows 3.1 16 Mb RAM, 17-25 Mb free disk space, 16-bit version also available.	Microsoft Office and Windows 95 compatible. Wizard guides users step-by-step in creating graphs. Edit graphs using graph toolbar. SigmaPlot's worksheet will be familiar to users of Excel and easy to learn.	From Jandel Statistical Software. (http://www.jandel.com). Has won the Users' Choice Award for the past five years from readers of Scientific Computing and Automation. Demonstration version available at site.
SigmaStat™	An expert system design provides guidance through statistical analysis. Options to explain test results. Can do statistical analysis through Excel spreadsheet running inside SigmaStat.	Independent and paired tests, 1/2/3-way ANOVA, non-parametric statistics, linear, multi-linear and polynomial regression, t-tests, descriptive statistics and mathematical transforms. 2-D and 3-D customisable graphs and plots.	One billion data points (16,384 columns and 65,436 rows). SigmaLink with SigmaPlot worksheet.	Windows 95 and NT 3.51 or higher with minimum 8 Mb RAM (16 Mb recommended), 11-16 Mb hard disk space. Windows 3.1 or higher with 16 Mb RAM, 15-20 Mb hard disk space and 20 Mb for swap file. 16 bit version available.	CD-ROM to help with statistics is available. Interface is Microsoft Office and Windows 95 compatible.	From Jandel Statistical Software (http://www.jandel.com). Demonstration available at web site.
SIMSTAT™	Offers output management features, its own scripting language, and computer assisted interviewing systems. High resolution graphics. Data editor has more than 47 operators and functions.	Descriptive statistics, paired and independent t-tests, one-way variance analysis, GLM ANOVA, correlation matrix, multiple regression, time-series, non-parametric analysis, bootstrap resampling, full analysis bootstrap.	No information available.	For DOS and Windows.	Has interactive tutorials with multi-media capability. Spreadsheet data editor.	From Provalis Research, Montreal. An evaluation version can be downloaded from http://ourworld.compuserve.com/homepages/simstat/. Registered SIMSTAT users are eligible for integration of Multivariate Statistical Program (MVSP) for PC compatibles that performs a variety of ordination and cluster analyses.

Product	Overview	Features	Data limits	Platform	Ease of use	Comments
SPSS™	Transformation, correlation, linear regression, non-parametric tests, ANOVA models	Comprehensive package. Automatic graphs, chart carousel, 3-D rotation, colour brushing, animation.	Virtually unlimited numbers of variables and cases. Imports data from ODBC.	Windows 95, 3.1 and NT, DOS, OS/2, Macintosh, Power Macintosh, UNIX.	Intuitive GUI.	From SPSS Inc (*http://www.spss.com*).
STATA™	Similar features to SAS and SPSS, but does not perform time series analyses. Extensive programming language. Spreadsheet editor.	ANOVA, bootstrapping, factor analysis, principal components, non-linear regression, survival analysis. Runs interactively or in batch mode.	Handles very large data sets.	Windows 95, DOS, Macintosh, UNIX.	Does not have a GUI but the program's syntax may be easier to learn and use than similar command-line driven packages. Net course available via e-mail for range of user abilities.	From STATA Corporation (*http://www.stata.com*). Monthly STATA technical bulletin aimed at data management, biostatistics and epidemiology communities.
Statview™	160 built-in arithmetic, algebraic, logical, string, statistical, or special purpose functions including IF, THEN, ELSE statements.	Customised graphics can be saved as templates.	Creates columns for over 20 different distributions and series.	Windows 95, NT, 3.1, Macintosh, Power Macintosh.	Graphical interface almost identical for Windows and Macintosh. On-line help includes hyper-text links. No need for programming.	From Abacus Concepts (*http://www.abacus.com*). Has been the leading statistical package for Macintosh which has English, French, German and Japanese versions.
SYSTAT™	Descriptive, correlation, log-linear models, non-parametric tests, regression, ANOVA, time series, matrix procedures, cluster analysis, programmable.	Comprehensive package. Displayed in 2-D, 3-D or pseudo 3-D, automatic graphs, 3-D rotation, maps.	Up to 32,000 columns of data.	Windows 3.1 or 95, DOS, Macintosh.	Requires programming knowledge.	From SPSS Inc (*http://www.spss.com*).

4.4. References and Bibliography

Albert, J. H. 1996. Bayesian Computation Using Minitab. Duxbury Press.

Anthony, J. 1996. Probability and Statistics for Engineers and Scientists. Publishing Co.

Chatfield, C. 1991. Statistics for Technology. Chapman and Hall.

Clark, G. M and D. Cooke. 1991. A Basic Course in Statistics. Edward Arnold.

Cressie, N. and J. J. Majure. 1997. Spatio-temporal statistical modelling of livestock waste in streams. Journal of Agricultural, Biological and Environmental Statistics, 2, 24-47.

Dominici, F., G. Partmigiani, K. H. Reckhow and R. L. Wolpert. 1997. Combining information from related regressions. Journal of Agricultural, Biological and Environmental Statistics, 2, 313-332.

Earickson, R. J. and J. M. Harlin. 1994. Geographic Measurement and Quantitative Analysis. Prentice Hall.

Gao, F., J. Sachs and W. J. Welch. 1996. Predicting urban ozone levels and trends with semi-parametric modelling. Journal of Agricultural, Biological and Environmental Statistics, 1, 404-425.

Gumpertz, M. L., J. M. Graham and J. B. Ristaino. 1997. Autologistic model of spatial pattern of phytophthora epidemic in bell pepper : effects of soil variables on disease presence. Journal of Agricultural, Biological and Environmental Statistics, 2, 131-156.

Hastie, T. J. and R. J. Tibshirani. 1990. Generalized Additive Models. Chapman and Hall.

Kuehl, R. O. 1994. Statistical Principles of Research Design and Analysis. Duxbury Press.

Ming-Hui Chen and J. J. Deely. 1996. Bayesian analysis for a constrained linear multiple regression problem for predicting the new crop of apples. Journal of Agricultural, Biological and Environmental Statistics, 1, 467-489.

Qi Jiang and P. A. Succup. 1996. A study of the specification of the log-log and log-additive models for the relationship between blood lead and environmental lead. Journal of Agricultural, Biological and Environmental Statistics, 1, 426-434.

Ross, P. W. (ed). 1996. The Handbook of Software for Engineers and Scientists. CRC Press, Boca Raton, Florida and IEEE Press.

SAS Institute. 1993. Hot spotting. European SAS System Journal, 6, 10-11.

Shaw, G. and D. Wheeler. 1994. Statistical Techniques in Geographical Analysis. Halsted Press.

Swan, A. R. H. and M. Sandiland. 1995. Introduction to Geological Data Analysis. Blackwell Science.

West, M. and J. Harrison. 1997. Bayesian Forecasting and Dynamic Models. Springer-Verlag.

5. NEURAL NETWORKS

5.1. Capabilities and Limitations

For many people, the term ‘neural networks’ is intimidating. It is associated with research into the most complex object in the known universe, the human brain, and so sounds unlikely to have any relevance to them. The good news is that modern artificial neural network packages are very powerful and easy to use. They make it unnecessary for you to get involved with the details of the technology[8]. More important, they incorporate some powerful capabilities from statistics to help you solve your end-to-end problem. This is a real advance from just a few years ago when neural networks packages were mostly for students and researchers.

What are neural networks?
Here is a rather general definition of what neural networks (NNs from now on) are:

> *a kind of artificial intelligence, based on simple ideas about how brains work, which learn from examples in the same way as a child, require no ‘programming’ in the conventional sense and are capable of generalisation about new unknown data.*

Here is a more technical definition. NNs are:

> ***general-purpose** computer programs, that are **black-box** in nature, can **learn** and **predict** and use **mappings** to relate one thing to another[9].*

Let’s expand on each point.

General Purpose: Unlike most computer programs, which are designed to do specific jobs, the tasks NNs perform are general. Thus, the same NN program may be applied to do very different jobs, for example, controlling a robot arm and classifying clouds. In this sense they are a little bit like spreadsheet programs, which are also used for a wide variety of jobs.

Black Box: A black box is something into which you cannot see. What this means here is that what is inside a NN – the set of internal parameters – is rather uninformative. They can learn, but they cannot inform you!

Learn and Predict: NNs have two modes: learning and predicting. In learning mode, they look at the inputs and outputs in your data and modify their internal parameters so as to ‘learn’ about the way the relationships vary. Having finished learning, they can be used to predict what the outputs would be for the inputs you supply. Warning: to operate properly, the inputs you supply in predict mode must be similar to those presented to the network in the learning mode.

Mappings: The relationships NNs learn are ‘mappings’ from some input space to an output space. For example, from handwriting to alphanumeric characters, or from recent stock prices to future stock prices, or from satellite measurements to crop health. The NN program is independent of what the mapping is about; that is why they find such a diversity of applications.

As the plural ‘networks’ implies, there are many different kinds; the list would take up many pages! In this chapter, we focus on the kinds likely to be useful for environmental applications, and ignore all the others. This means we concentrate on the Multi-layer perceptron, the Radial Basis Function network and the Kohonen network. We will explain what these are later.

[8] Of course, knowing the details helps get the best results out of any technology!

[9] There are special neural network types that do not fit all the terms in this definition.

To summarise this section:

> *Neural networks are tools for building models from data. They can be applied whenever there is an unknown relationship between inputs and outputs, and there is an adequate supply of data illustrating the relationship.*

How are neural networks different from conventional statistics?
Some say that NNs are just a different way of doing statistics. That point of view is debated, but it is certainly true that the statistical basis of NNs is becoming clear, and the two domains are beginning to move together. In common with statistics, the role of NNs is to discover and quantify relationships among variables. Statistics offers a strong but rigid analysis framework to do this, while NNs offer a more flexible framework.

The consequences of this are:
- NNs are easier to use than statistics, in that it is not necessary to understand the data distributions to use them.
- NNs rely on the data containing all the information. As we explored in Chapter 4, the form of the data distribution is something that the statistical analyst can specify to help get the information out of the data. This can be risky; but it can also be very powerful. With NNs, there are very few ways to insert your own knowledge (or bias) into the solution.

Another way of expressing the difference is to say that NNs capture the average behaviour of the data, while statistics works on this and also the internal relationships among the variables.

Why is it important? Or rather when is it important?
If you need to gain understanding about why a system behaves the way it does, NNs are not the best tool. If you just need something that can make a reasonably good prediction, then NNs can be quite helpful. For example, if you have data on population density, traffic patterns, rainfall, and the distribution of smog in a valley, and you want to predict the effect of changes in smog as a function of the first three, you may be unwilling or unable to model the physics and chemistry of the relationships. You may prefer to train a neural network and see what it predicts. Perhaps you may like to do both.

NNs are already being used in many international companies and institutions for forecasting as well as other predictive and classifying tasks. This is because they are recognised to have several key advantages. NNs can handle:
- complex multi-variate relationships
- non deterministic problems
- non-linear problems
- noisy data.

In addition, they offer:
- fast speed of analysis
- objective viewpoint
- ability to generalise
- extrapolation beyond initial data range
- simple and quick update process (no expert required).

How do I decide whether to use a statistical or neural network approach?
Begin by visualising your data. If you can develop some confidence that the relationships among the variables are linear (straight-line) and the underlying data distributions are canonical (normal, exponential, poisson) then use statistical tools. If you can transform your variables so that you have linearity, then again use statistical tools. However if the relationships are non-linear, or there is reason to suspect that the distributions may be rather odd, then consider a NN approach.

However, there is another criterion that must be met for NNs to be of use to you. They like lots of training data, which must be well-distributed. Can you provide it? This is important, so we will discuss and illustrate it now. The reason NNs require so much training data is that they rely on the data containing all of the information. This is the penalty paid for making few assumptions about the nature of the data. So how much data is necessary? The driver is the number of variables, including nuisance variables (see Box 5.1). Variables are the problem inputs, the factors that affect the system output(s). If there is a single variable influencing the output, then the data you feed the NN must have a sufficient number of examples of that relationship to allow the network to learn the important features of the relationship. If the relationship is linear, for example, you would want data at the minimum, maximum, and several points in between. More if the relationship is not linear, and lots more if the data is noisy.

On the other hand, if you had one variable that needed sampling at 10 points, and a second variable also needing 10 points, there is a matrix of possibilities with 100 combinations. If they are all possible, you will need at least one data sample for each of them (preferably many more). Yet again if you had six variables, you would need at least one million samples. This is called 'the curse of dimensionality'; the more variables you have, the more data you need. It's often not as bad as that, because not all of the possibilities in the matrix are important. Some variables may have no effect on the output, and if there are relationships among certain variables, some of these can be eliminated.

So before adopting a NN approach, consider how much data you will need and how much you can afford. Box 5.1 illustrates what we have been discussing in the context of a hypothetical waste-treatment plant.

Box 5.1: Estimating the data requirements for a neural network

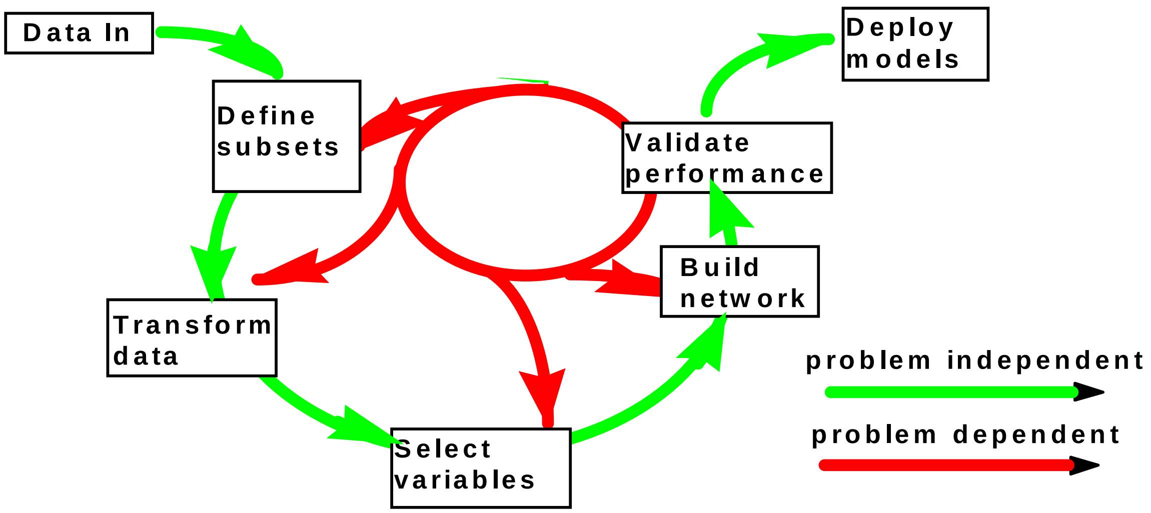
> Waste treatment plant operators want to know how plant factors affect the discharge quality, so they can optimise performance. The factors they can control include the input mixture, the quantity of treatment chemicals they use, and the treatment time. Other factors that affect the output, but which are not under their control (nuisance variables) include the amount of water in the input mixture, and the temperature of the input mix (which is controlled by the weather). They treat waste in batches, at an average rate of 10 per day.
>
> They have enough experience that they can work out rough data requirements. They classify the input mixture in 40 different categories, all of which result in different discharge quality levels. They think that five different levels of treatment chemicals will be enough, and five different treatment times. So there are 25 x 40 = 1000 combinations. The amount of water in the input mixture is easy to estimate from the weight, and it makes a big difference to the plant efficiency. So 10 different levels here, and then there is the weather; say five; 50,000 combinations in all.
>
> Looking for ways to reduce this, they consider some rules of thumb that relate treatment chemicals to input mixture. So this factor could be reduced from five to three (what the rule of thumb indicates and the next higher and lower). The operators are certain that short treatment times in winter will not be adequate and long treatment times in the summer are not necessary, so the range of that variable can be cut in half. With these and other trimmings, the number of samples is cut to 10,000, or 1000 days' data collection. By running the data collection program at five similar plants simultaneously, enough information could be collected in one year's operations. A careful cost-benefit analysis indicates that if a 3% improvement in efficiency is achieved, the data collection program will pay for itself in one year.

To summarise the difference between the NN approach and the statistical approach, statistics provides a very strong framework for data analysis that is appropriate when you have confidence that the form of the data distributions is known. NNs are more flexible and easier to use than statistics, because they eliminate the need to understand the data distributions. But there is the penalty that it is harder to validate a NN. The next section introduces the relevant capabilities of NNs. It goes on to explain how to gain confidence that a NN is behaving sensibly – how to know when it should be trusted and when not.

5.1.1. Key Capabilities

Neural network vendors have taken seriously the need to support users in the end-to-end analysis of their data analysis problems. Figure 5.1 illustrates the point that training a NN (model building) is only one of six steps on the way to realising a successful neural model. The figure makes the point that model iteration is often necessary; the lessons learned in one version of the model lead to refinements and improvements. The amount of iteration of course depends on how complex the problem is and how important the result is. Stock market analysts go around the loop many times.

Figure 5.1: Neural modelling cycle[10]



Each of the boxes in the diagram corresponds to an important capability which a NN tool should offer:

- defining data subsets
- transforming the data appropriately
- selecting suitable variables
- building the neural network
- validating performance
- deployment.

As the figure illustrates, some of the steps are problem independent – this means a suitable tool can perform them automatically. The part that is problem dependent is the iteration: finding things that you can do to improve the performance of the network. These can involve introducing special knowledge that you have about the problem, for example constraining a part of the solution to fit a model, such as linearity. Each of the NN tool capabilities are now described.

Defining data subsets

Although it seems trivial, the selection of appropriate data subsets is actually the most important step for ensuring the validity of the network. Three subsets are required: one for training the NN (a large one), another for testing the network while it's being trained – so you know when you are finished, and a further one for validating the trained network. It is important that the validation

[10] © Scientific Computers Limited (used with permission). As a point of interest, the figure is typical of any data modelling cycle, not just neural models

dataset be independent of the training and test datasets, otherwise the performance estimates will be optimistic.

The simplest data selection strategy is to draw random subsets from the complete dataset. This is appropriate if the dataset is large enough. If the dataset size is small, say less than a few hundred samples, then simply reserving a random 10% for test and 10% for validation might be inappropriate; there is a chance that the few 10s of samples in each set will be a poor selection, thus biasing the performance measures. In these cases a ‘leave one out’ strategy, also known as ‘jack-knife’ may be appropriate: With a dataset of size 100, use 99 to train the network, and test on the remaining one. Then do the same again 99 times, leaving out each one in turn.

Transforming the data appropriately

NNs require numerical data, and they work best if the ranges of the variables are similar, for example in the range 1 to 1. Performance is usually best if the distribution of variables is fairly even over the range. If your data contains non-numerical data, or the variables have a variety of different ranges, or if the distribution of the data is very skewed, you will need to find suitable transformations that make your data easier for the network to handle. For example, if a variable’s values lie in the range 0.0 to 10.0, but 90% of the values are in the range 0.0 to 1.0, you will improve performance by appropriately transforming the variable (Log_{10} (1 + x) might be a suitable transformation).

Selecting suitable variables

Quite often, datasets contain many variables, of which some are irrelevant and others are redundant. As indicated earlier, the smaller the number of variables you can work with, the less data you need. Another way of looking at this truth is if fewer variables are sufficient to define the relationship, then for the same data you can get a better NN. The job of selecting appropriate variables, like that of identifying appropriate transformations, is one that the more user-friendly NN packages manage for you, at least to a first approximation.

Building the neural network

NNs have a number of parameters that can affect the performance (number of nodes, training rate, momentum[11], for example). In the past, finding a suitable network configuration was a matter of trial and error. For each set of configuration parameters, a tedious cycle of train, test, and validate was necessary. In fact it still is, but now with at least some packages, automatic configuration procedures help you through this phase.

Validating performance

In this step the reserved data from the first step, that the network has never been shown, is used to get an honest calibration of performance. You need to know and be able to convince others that the flexibility of the NN has not been wasted or over-used. If the network is not properly tuned, the performance will be poor, and if it’s over-tuned, it will follow statistical fluctuations in the training data. The quality of the validation data, and its independence from the training data, is therefore crucial. The validation dataset must be large enough to provide a good performance estimate over the full range of the network’s expected operation.

Having measured the network’s performance, there is a choice: deploy the network (it’s good enough) or iterate to try and improve performance. Usually it’s a good idea to iterate at least a few times, to ensure that the solution you have found is reasonably stable. How much iteration is a matter of judgement: how valuable are the potential improvements versus the extra effort involved. Be careful; it is easy to fool yourself chasing statistical fluctuations. A good rule of thumb is the 1/sqrt (N) rule. If N is the size your validation dataset, 1/sqrt(N) is a percentage that indicates the statistical quality of your data – the lower the number, the more accurate. Changes in performance below the level indicated in this way are almost certainly due to chance. There is a

[11] The parameters that affect performance are different for different network architectures; we don’t go into these details here

slightly subtle but very important principle that is often overlooked at this point. If any of your validation data are used to refine the NN structure, then they lose their validity; effectively they become part of the test data. So if you are going to iterate your solution, you need to hold some 'pristine' validation data aside until the very end, to get an objective performance estimate.

Deployment
Once you have established good performance with your network, you may want to take it out of your PC and put it to work in another context. Most packages allow you to export the model as C/C++ code, so that it can be used as a resource by other software. This is a very important capability, because when the network has been trained and is performing acceptably, what you often need to do is get it out of the interactive environment in which it has been constructed and integrate it into the information system that needs it. For example, a network trained to forecast air pollution might be used as a component of a real-time air quality control centre. The operations staff do not want to know the details of the network or see anything about its graphical user interface; they just want it to deliver its results. A software engineer can quickly do this integration, once the network code is available.

5.1.2. Varieties of Neural Networks

One of the most useful ways to categorise networks is by how they learn. Supervised networks demand that you provide an example of the output(s) required for each set of inputs, while unsupervised networks train themselves to classify inputs into groups, or clusters. For most applications, it is preferable to train a supervised network, because of the control this gives you over the output categories. Of course, this requires suitable training data. The next few paragraphs and Table 5.1 introduce the three types of NNs most likely to be useful for environmental applications. As a prelude to the main discussion of this section, the paragraphs below may be helpful in forming more concrete ideas about NNs. NNs were first described as innovative computational tools for optimising environmental data processing by Schmuller (1990). Schmuller provided a tutorial introduction to NNs that is particularly useful for those with little or no background in the field. He illustrated a simple neural network of the type shown in Figure 5.2 with input units i, that takes input data X_i and multiplies them by the weigh of the interconnections w_{ij}. The sum (A_j) of these products is then calculated producing activation values at different output unit *j*.

$$A_j = \sum X_i\, w_{ij} \quad i = 1, 2, 3, 4, 5 \ldots\ldots\ldots 15$$

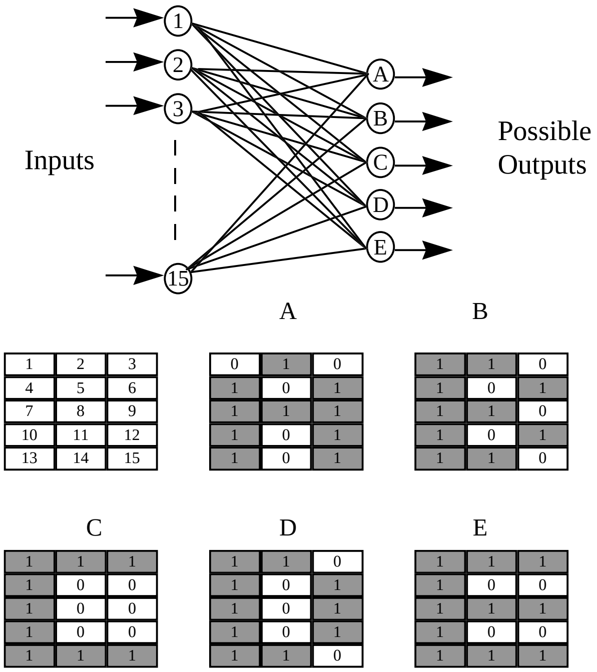
In the output layer if the activation value exceeds some pre-set 'threshold' value, the unit provides an answer that in the simplest case is either 1 or 0. In the example the 15 input cells correspond to black and white colours, or 1 and 0 values, the objective of the NN is to recognise the first five letters of the alphabet according to the value of these cells. For the letter A for instance the distribution of the 1,0 values is: [0,1,0,1,0,1,1,1,1,1,0,1,1,0,1]. Each of the five output units is then assigned a letter. The first output-layer unit would recognise only the letter A giving the value 1 if this pattern is present or 0 if is not. The way each output-layer unit recognises the letter is because a set of weights is attributed to it according to the distribution of the 0-1 values in the 15 cell matrix grid.

The type of network illustrated in Figure 5.2 is called a 'Perceptron'. The most popular supervised network is the 'multi-layer perceptron' (MLP). Rather than just having the input and output layers shown on the left and right of the figure, MLPs have one or more 'hidden layers' lying in between. Until recently, it was said that 90% of the world's NN applications were based on the MLP. MLPs are sometimes also called 'back-propagation' networks, a term that refers to the algorithm used to train them.

Table 5.1: Strengths and weaknesses of the major NN types for environmental applications

Type / Applications	Strengths	Weaknesses
Multi-Layer Perceptron	supervised ease of use widely supported	slow training speed deteriorates rapidly with problem complexity no possibility to indicate when outside training domain
Radial Basis Function	supervised rapid training ability to indicate when outside training domain reasonably fast in prediction mode	speed deteriorates with number of inputs, but not as fast as MLP availability improving, but not yet widespread
Kohonen	unsupervised widely available reasonably fast in prediction mode, like RBF	slow training need to specify number of data categories in advance

Figure 5.2: A simple neural network for character recognition



1	2	3
4	5	6
7	8	9
10	11	12
13	14	15

A

0	1	0
1	0	1
1	1	1
1	0	1
1	0	1

B

1	1	0
1	0	1
1	1	0
1	0	1
1	1	0

C

1	1	1
1	0	0
1	0	0
1	0	0
1	1	1

D

1	1	0
1	0	1
1	0	1
1	0	1
1	1	0

E

1	1	1
1	0	0
1	1	1
1	0	0
1	1	1

Another type of supervised NN that is gaining popularity is the ‘radial basis function (RBF) network. The difference between an MLP and an RBF is that the MLP generalises globally, whereas the RBF generalises locally. What this means is that when you query an MLP, every training example it has ever been shown will affect the output, whereas when you query an RBF, only those training examples that are similar to the query will affect the output. This affects not just what the output will be, but also how quickly it can be evaluated – the RBF is faster except for very small networks.

The leading example of an unsupervised NN is the ‘Kohonen”, named after its inventor. A Kohonen network can be very helpful, in that it can take a high-dimensional input (many variables) and find a way to reduce the number of variables to two. Another way a Kohonen is used is by training it to organise the data into clusters, then a person labels the clusters with class names. After that the Kohonen can be used as a classifier.

5.1.3. Limitations and Likely Evolution – the Authors' View

After much rapid progress in the 1980s, the first half of the 1990s has seen consolidation and incremental advancement. The recent arrival of packages that deal with the end-to-end problem is very encouraging, but users would welcome even more capability of this type. New architectures, such as the 'growing neural gas' of the University of Bochum, *(http://www.neuroinformatik.ruhr-uni-bochum.de/ini/VDM/research/gsn/DemoGNG/GNG)* offer networks that adapt themselves even more to the characteristics of the data. We can look forward to the day when ready-packaged networks will be able to grapple with large problems in a highly intelligent, automated way. In particular, we see two needs:

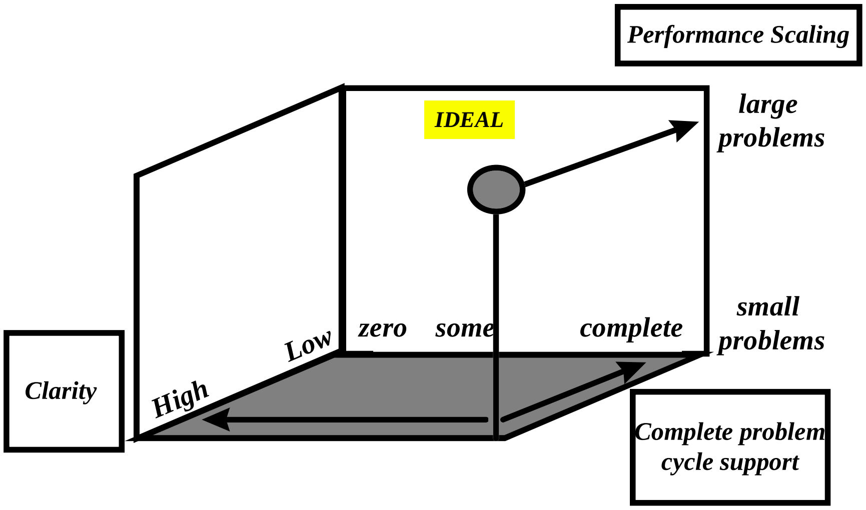
- to deliver user-meaningful explanations of the relationships found in the data
- to detect and warn the user when network responses are unreliable.

Hardware implementations of NNs are gaining power and relevance. What this means for the environment is that we can expect dramatically better automatic feature extraction capabilities, suitable for image-based environmental monitoring and other large volume sets such as video cameras and satellites products. Another important emerging
trend in NNs is closer integration with statistical techniques. In a way, many varieties of NNs can be viewed as just more flexible data fitting tools. Statistical techniques can be applied to estimate the uncertainties in the weight parameters of these networks; analogous to the standard error bars that statistical techniques give for linear regression parameters. Networks enhanced in this way can give an indication of the accuracy of the results they return.

5.2. Representative Neural Network Tools

Figure 5.3 shows three of the factors that are most important in selecting an appropriate NN tool.

Figure 5.3: Neural network package selection factors



An 'ideal' tool would be able to handle as large a problem possible, provide support for the complete problem cycle (as described earlier), and would be so helpful and clear that we would never get confused or make mistakes using it. Of course, there are other important factors as well:

- the number of different network architectures supported
- the degree of control over network parameters

cost and availability.

In the next sections we will describe two commercial NN tools, to illustrate the points we have been making. The first is a simple Excel-based NN and the second is a hybrid NN and neuro-fuzzy expert system. A selection of other NN tools are then listed in Section 5.2.3.

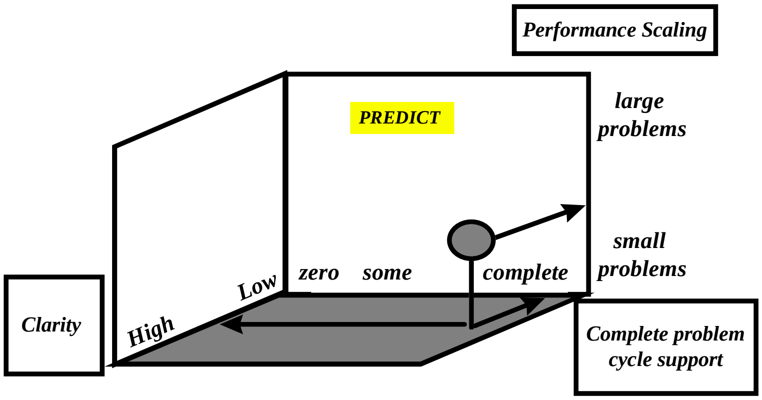
5.2.1. NeuralWorks Predict™ Professional – PC Excel version

Predict™ is a tool for developing and deploying neural network applications in forecasting, modelling and classification. It is designed primarily for people without a background in neural technology, who nevertheless wish to explore and perhaps deploy a NN-based solution. Three versions are available: Lite, Standard and Professional. All run in a PC-Excel environment. As Figure 5.4 illustrates, one of Predict's strong features is that it addresses all the stages of the problem cycle, as described earlier. However, being embedded in Excel, it is limited to the maximum size of a spreadsheet (and in fact runs out of performance before then).

We ranked Predict in the middle region of the clarity axis. The reason is that beneath its surface is NeuralWorks Professional, one of the most comprehensive research tools for NNs. From time-to-time menus pop up with a bewildering array of options, and it would be rather easy to become confused and make mistakes without a close eye on the Predict manual. That said, what Predict achieves is most impressive – a high degree of automation of NN generation process. Predict has a wealth of features and options; so many that it is impossible to cover them all in the short space available here. The next sections pick out some of the features of this package that seem most likely to be interesting to environmental workers.

Ease of use
Predict adds an additional column to the Microsoft Excel menu bar. Within this menu, Predict caters for different levels of expertise. The Basic mode for the least experienced is fully automated; users are prompted only for data selection and type of problem, and are guided by helpful dialogues. Advanced and Expert modes give users increasingly more control over the parameter selection and details of the internal algorithm. It is possible to be doing useful work with Predict in less than two hours after opening the package. However, several days should be set aside to familiarise oneself with the more advanced capabilities of the product.

Figure 5.4: NeuralWorks Predict™ selection factor ratings



Data input, variable selection and transformation
Data is entered through the Excel spreadsheet. The columns that represent input and output are simply selected from the spreadsheet. An automatic variable selection routine (based on a genetic algorithm) is used to select the minimal set of inputs. Missing data is handled in a variety of ways. Too many missing values cause the field to be rejected; in other cases, values are assigned or flagged. Predict guides the user in transforming data into a form suitable for network training. It handles enumerated variables, such as postal codes or marital status, through linear or one-of-n encoding. Continuous variables are mapped to the interval [-1,+1] with histogram-based density equalisation.

Network training, validation and deployment
Network training is supported by two non-linear feed-forward constructive algorithms. One is optimised for low-noise and the other for noisy data. Both automatically determine optimal learning rates and architectures. Network validation is achieved by running the network on an independent dataset. When the final network is finished and working properly, the user can choose to export the entire application to portable C-code, Visual Basic, FORTRAN or DLL. The exported code can be integrated into an existing application or deployed as a standalone application.

Limitations
Predict does not have its own graphics, and it offers only a limited set of performance metrics, but the graphics and statistical capabilities of Excel can easily be invoked as required. The increased automation of the processing means the product is not fully customisable to the user's needs. Predict only supports the MLP architecture.

Documentation and support
The on-line help is excellent, explaining not just what the product does, but also why it does it. Telephone technical support is provided by NeuralWare's product support specialists. On-site product support is available through NeuralWare's global network of distributors and system integrators. Training is available through regularly scheduled training courses organised by international distributors. On-site training is also available, although your local distributor should be contacted through NeuralWare.

Cost and availability
NeuralWorks Predict is available from NeuralWare Inc (*http://neuralware.com*). It requires Microsoft Excel 4, 5 or 7. On Intel processors, Predict requires Windows 3.1, Windows 95 or NT. It needs a minimum of 8 Mb RAM and 6 Mb disk space. Table 5.2 shows the cost factor rankings for Predict, using the categories defined in Table 2.1.

Table 5.2: Cost factor rankings for NeuralWorks Predict™

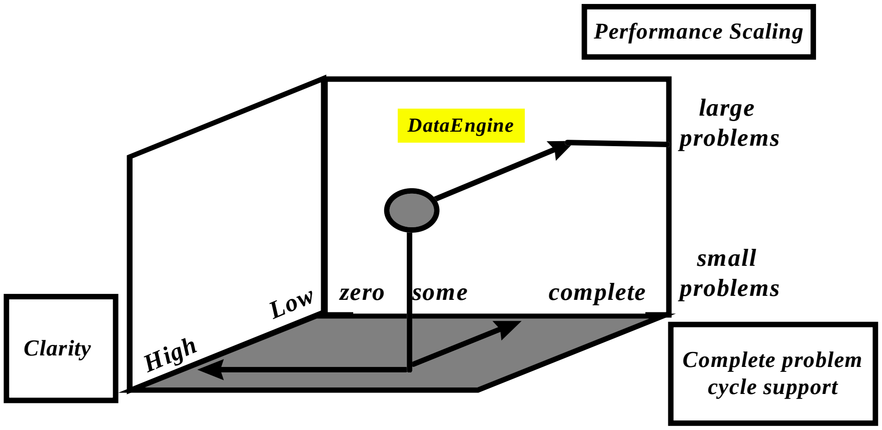
learning	installation	application	price
B-C hours to days	A less than 10 minutes	B-C hours to days	F Professional version

5.2.2. DataEngine™ _ – a hybrid NN and neuro-fuzzy expert system

DataEngine from MIT GmbH, combines conventional data analysis methods with the intelligent technologies of Fuzzy Rule Based systems, Neural Networks and NeuroFuzzy systems[12] , making it a hybrid NN-expert system tool. A similar package (NEUframe) is featured in Chapter 6. As Figure 5.5 illustrates, we rated DataEngine higher than Predict on the clarity axis. The reason for this is the statistical processing and visualisation tools it offers. Many NN/fuzzy tools neglect the importance of quantifying, visualising and understanding both the data and the developed solution. DataEngine applications usually involve some pre-processing and histogram/time series visualisation before the data is passed to the neural/fuzzy network. Like Predict, DataEngine offers good support for the complete problem lifecycle; however it misses some aspects that other packages offer, such as input selection routines and automatic determination of which inputs are relevant. Because it is not limited by a spreadsheet, DataEngine can handle larger problems than Predict. The following paragraphs give an expanded discussion of the selection factors, and explore some other features of DataEngine likely to be interesting to environmental workers.

[12] See Chapter 6.

Figure 5.5: DataEngine™ selection factor ratings



Ease of use
DataEngine has a graphical interface with functions to automate pre-processing, data analysis and visualisation. The selection menus and graphics of DataEngine assist you to design a network and view the current status of your network. The graphical macro language is easy to use and enables automation of analysis procedures. The data graphing capabilities include time series, 2-D and 3-D line plots, scatter plots, normal or stacked bars and pie plots. Charts can be dynamically linked to data using the macro language.

Data input, variable selection and transformation
Data is handled in various file formats. Excel spreadsheets can be imported as can other ASCII files. Spreadsheet functions can be edited to create and visualise the data. DataEngine also supports interfaces to data acquisition boards. The data editor employs a broad range of functions for the pre- and post-processing of data. These include arithmetic functions, logarithmics, hyperbolics, scaling, linear regression, correlation, and descriptive statistics such as variance and standard deviation. It can also handle missing values in various ways including replacement by a constant or interpolation. The signal processing module of DataEngine has an IIR-digital filter with options for high-pass, low-pass, band pass and band stop, a Fast Fourier transformation (FFT), inverse FFT and data smoothing capability.

Network training
A limited range of neural and fuzzy networks are supported inside DataEngine – with MLP and Kohonen networks as the main supervised and unsupervised techniques, respectively. MLP training is achieved using back propagation and quick-prop, although none of the more advanced structuring techniques (such as weight elimination, second order training, and regularisation)[13] are currently supported. This contrasts with Predict, which has a much wider range of training/network options, but is more difficult to use and has limited graphical/statistical support.

Extendability
The graphical macro language provides user-defined function blocks for adding specialised analysis techniques or database interfaces. These are based on Windows Dynamic Link Libraries (DLL). The DataEngine ADL (Application Development Library), a stand-alone product, can be combined with DataEngine. This is a library allowing users to integrate solutions developed in DataEngine, into their existing software environment. It is offered as a DLL for Windows or as a C++ library for various compilers and platforms. It will require commitment to exploit these capabilities.

[13] These advanced training methods improve the performance of back-propagation networks, at the cost of longer training time.

LabView™, which is a software environment for controlling computer-compatible instruments such as weather stations and data logging devices, has an interface to DataEngine. This enables the combined system to be used either as a simulation environment or as a tool to control sensors in real-time.

Documentation and support

A telephone hot line is available for trouble-shooting. Training courses are held either at the distributor's premises or on-site, as well as at MIT offices. The courses are tailored to meet the knowledge and requirements of the user. The implementation of turn key solutions in various areas is also offered. Support is also available via annual meetings at EUFIT and various workshops that are part of the ERUDIT EU network of excellence.

Cost and availability

DataEngine is supplier through distributors for MIT GmbH *(http://www.mitgmbh.de)* and is available for Windows 3.1, Windows 95 or NT, requiring a minimum 386, 33 MHz, 8 Mb RAM. It also is available on most varieties of UNIX workstations. A full licence, an educational licence and a limited full licence are available for both the PC and workstation versions. Table 5.3 shows the cost factor rankings for DataEngine using the categories defined in Table 2.1.

Table 5.3: Cost factor rankings for DataEngine™

learning	installation	application	price
B-C hours to days	A less than 10 minutes	B-C hours to days	E-F PC / workstation + annual licences

5.2.3. Other NN Products

Table 5.4 provides summary details of some of the other NN tools that are on the market today.

5.3. Environmental Examples

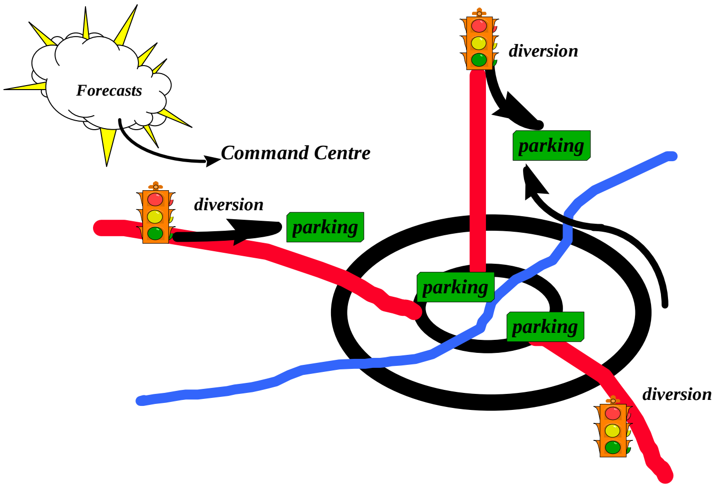
Neural networks have long been recognised as powerful and economical tools for solving a large variety of problems in science and engineering disciplines. In recent years their utilisation is fast expanding into more environmental fields, considerably aided by the growing number of PC-based tools. To support this, the literature on NNs is enormous, consisting of books, journals, conference proceedings and magazines. Two of the best magazines include AI Expert and PC-AI, both are relatively non-technical and aimed at artificial intelligence technology for small computers. The most prestigious journals are the 'IEEE Transactions on Neural Networks' and 'Neural Networks', which is the official journal of the International Neural Network Society; both periodicals are technical journals devoted exclusively to current research on neural networks. A selection of introductory text books are included in Section 4.4. As with other CIPTs most of the NN vendors provide detailed information about both the technology and their products on their web-sites and Neuronet aims to provide an on-line repository of relevant information and resources *(http://www.neuronet.ph.kcl.ac.uk/neuronet/software/software.html)*. Another useful web-site for technical and applications information can be found at the Pacific Northwest National Laboratory (PNNL), operated by the Battelle Memorial Institute *(http://www.emsl.pnl.gov:2080/docs/cie/neural/neural.homepage.html)*.

In recent years the capabilities of NNs to forecast environmental events such as ocean currents or human activities such as road traffic volumes has met with varying degrees of success (Smith et al, 1995; Baptist et al, 1994). Baptist et al used NNs to process ocean and atmospheric data sets that contained ocean surface temperature, air pressure and wind data from 1884 to 1994 to forecast El Niño's current events. The tropical Pacific Ocean is affected by the El Niño phenomenon in terms of significant climactic changes, rainfall patterns, storm frequency and intensity. Fisheries off the coast of much of the Americas are also affected by this current. NNs proved to offer quite reliable analytical forecasting capabilities to predict El Niño's current changes.

Before we go on to describe some case studies using NN, particularly in traffic management, waste site screening and air pollution, we have included an hypothetical case in Box 5.2, which illustrates the potential of NN[14] technologies for the creation of an advanced system that connects traffic patterns, driver response to forecasts, and air quality.

Box 5.2: Traffic management for air quality control

The negative relationship between traffic levels and air quality is well established (see for example 'The Citizens Network – fulfilling the potential of public passenger transport in Europe' 1996 EC green paper). Less well understood is how best to respond to temporary air quality crises. As illustrated in the figure, the flow of traffic into the hypothetical city is monitored together with air quality, and when air quality deteriorates, traffic is re-routed, to alternative parking lots within the centre or to out-of-town parking served by public transport. This system is re-active and inconveniences citizens; one would like to anticipate the problem and alert people so that they can change their plans in a timely fashion. NN technology could be employed to help make this possible. Here is a suggestion how it could be done.



The relationship between air pollution, traffic, and weather is complex and time-delayed. Configure a NN to learn to predict the present levels of air pollution as a function of earlier weather conditions, time of day, and traffic load on each artery. Use a NN that can perform time-series forecasting, based on these three variables. Configure a second network to predict traffic load on each artery, as a function of time, for each day of the year. When the networks' performance is good enough, begin issuing predictions early in the day. Train a third network to predict traffic load in the presence of predictions, and when this is good enough, substitute it for the second network. Estimate the economic savings, perhaps by surveying citizens.

Conventional traffic control techniques interpret traffic data by way of threshold tables. Accordingly, the classification of a specific state depends on the threshold interval. Problems arise, when different input data require different threshold tables, which have to be aggregated in order to give an overall traffic state classification. This leads to at least two problems: Crisp thresholds do not always represent exactly experts' judgements and simultaneous interpretation of various input data is difficult by combining the respective threshold tables. Kirschfink and Weber (1993) have developed a rule-based fuzzy system that uses linguistic variables instead of crisp threshold tables. The aggregation of all relevant input variables is performed by applying a fuzzy rule base instead of several threshold tables. This system has been developed with DataEngine. By using a rule-based fuzzy system, operator's knowledge and experience is much easier to implement and process than by previously used threshold tables. This leads to a much higher acceptance of the proposed traffic state classification system. A special development of DataEngine ADL allows the

[14] As Box 4.2 indicates, neural networks are not the only way to create such systems.

user at a traffic control centre to modify linguistic variables at run-time, so that dynamic changes can be performed immediately.

In another case study in the Intelligent Transport Systems (ITS) sector, NNs were evaluated as a tool for forecasting traffic volumes at two sites along the U.S. Capital Beltway in northern Virginia (Smith et al, 1995). In particular, an attempt was made to use traffic volume forecasts to govern proactive measures to alleviate congestion. Four models were applied: historical average, time series, NN and non-parametric regression models. The non-parametric regression model significantly outperformed the others. The study concluded that non-parametric regression offered other advantages such as portability, accuracy and easy deployment in the environmental field. The use of NNs in ITS was, however, found appropriate by the Transportation Research Board (1994) where comparisons were made with fuzzy set theory to improve freeway incident detection and hydrodynamic models for a freeway carrying morning commuters.

NN applications in environmental site screening and remediation appear quite established and well documented (Dabri et al, 1993). Traditionally, waste site characterisations in terms of rates and flow around and below the site required chemical analysis of physical samples extracted by drilling several bore holes and processing the data. This operation is however very costly and time-consuming. By training NNs with data collected in prior investigations, environmental scientists in the U.S. showed that satisfactory generalisation capabilities can be achieved. Alternatively, when uncertainties are such that NNs cannot be used alone, a limited number of bore sites to map and analyse waste plumes is recommended to complement the NN. One of the obvious evaluations in characterising waste site surroundings is the determination of optimum well placement.

NNs were also applied for analysing spatially distributed radio-ecological monitoring data collected in 1995 from the Chernobyl fallout (Kanevskij, 1995). Similarly Radknight et al (1997) used NNs to model the interactions that occur between ozone pollution, climatic conditions and the sensitivity of crops and other plants to ozone. They present a number of generic methods for analysis and modelling of relevance to non-linear variables. Multi-layer perceptron NNs were used to model data from a number of sources and analysis of the trained optimised models determined the accuracy of the model's predictions. Comparison is made of the accuracy of predictions for a number of modelling approaches. They show that the NN approach is more accurate than other methods and that the use of principal components analysis on the inputs can further improve the model. The relative importance of the causal agents in the model is established by summing absolute weight values, then a range of conditions is applied to the model to allow predictions to be made about the non-linear influences of the individual principal inputs and of combinations of two inputs viewed as a 3-D graph. Equations were synthesised from the NN to represent the model in an explicit mathematical form.

Other NN applications in the environmental field include the prediction of atmospheric pollutant concentrations due to fossil fuel electric power generation (Hashem, 1995), recognition of individual chemicals in complex mixtures using multi-spectral laser remote sensing systems (Wagner et al, 1994), the automatic identification of contaminants in field investigations using compact, portable systems with NNs implemented in software (Keller et al, 1994), and the simulation of ground water flow and transport for the optimisation of ground water remediation strategies (Rogers, 1992).

Table 5.4: Other neural network products

Product	Overview	Features	Data limits	Platform	Ease of use	Comments
Braincel™	A PC software for generating NNs. Combines with Excel and Lotus 1-2-3 to create 'intelligent spreadsheets'. Can make use of the spreadsheets functions and plotting and charting capabilities.	Backpropagation or 'backpercolation' algorithm (which is 100 times faster in some problems) allowing for high-speed, real-time modelling. Adds 20 new functions which can be run within the spreadsheet.	Excel limitations of 16,384 rows and 255 columns, 277 macro sheet functions and 246 worksheet functions. Lotus allows for 8192 rows and 255 columns, 108 macro sheet functions and 88 worksheet functions.	Windows running Excel or Lotus 1-2-3.	Has the interface of Excel or Lotus and is therefore familiar to end-users. Advanced users can fine tune for better performance.	From Promised Land Technologies (*http://promland.com*). The package can be run in beginners' mode or in the advanced professional mode. An automated approach, but does allow for text values and blank cells.
Cortex-Pro™	A general purpose NN simulation environment. The interpreted NN language allows control over simulations.	Built-in learning algorithms include backpropagation and Kohonen learning. Graphic commands provide illustrations and diagrams.	No information.	An IBM-PC compatible 386 SX processor or higher. Maths co-processor highly recommended. 4 Mb RAM. DOS version 3 or higher. Requires a Watcom 32 C (or C++) compiler.	Requires programming in C/C++. A true development environment therefore more suitable for application developers.	From Unistat Ltd (*http://www.unistat.com*). Free demonstration disk available from unistat@unistat.demon.co.uk or by downloading from (*http://www.neuronet.ph.kcl.ac.uk/neuronet/software/cortex/demo.html*).
Cortex-Pro™ / NeuralNet	A NN add-on module for the DADiSP software.	Backpropagation. Data can be pre-processed through the hundreds of analysis functions available in the integrated worksheet.	Unlimited input and output variables. Unlimited number of runs.	No information.	Menu-driven network design makes it easy to use.	From DSP Development Corporation (*http://www.dadisp.com*).
ECANSE™	A general visual programming tool with basic modules that can be combined to design a software solution by selecting the modules on a graphical interface.	Modules include data interfaces, signal generators, mathematical and statistical functions, script language, graphical displays and NNs, Fuzzy Logic and Genetic Algorithms. Tools for I/O, graphical visualisation, performance analysis and optimisation.	No information.	X-UNIX platforms, 16 Mb RAM, 30 Mb for User version and 60 Mb and AT&T C++ (Cfont 3.x) compliant Compiler for Programmers version. Windows NT – User version, Programmer's version, Run-time version.	Graphical user interface with visual programming environment. A scripting language is required to automate or modify some processes.	From Siemens Nixdorf Evaluation copy available from the ECANSE web site. (*http://www.siemens.at/ecanse/release.html*).
MATLAB	A toolbox providing a complete NN engineering environment within MATLAB. Architectures, learning rules, or transfer functions can be customised or new ones added without FORTRAN or C code.	Supervised network paradigms, unsupervised networks, competitive, limit, linear and sigmoid transfer functions, dynamic simulation with SIMULINK, portable C code generation through Real-Time Workshop and 15 proven network architectures and learning rules.	The toolbox imposes artificial limits on network size or connectivity so there are no restrictions on the number of neurons in a layer or on the type of transfer function.	Windows, Macintosh, VAX/VMS, most UNIX workstations, Convex, Cray and others.	Programming expertise is not required. A user's guide introduces neural concepts and reinforces them with numerous examples and a complete reference section.	Mathworks Inc (*http://mathworks.com/neural.html*). Price depends on platform but substantial discounts are available for educational establishments. Non-PC / non-Macintosh platforms are eligible for automatic upgrades. PC and Macintosh upgrade fee depends on size of upgrade.

Product	Overview	Features	Data limits	Platform	Ease of use	Comments
NEUFrame Professional ™	A simple, component based, point-and-click system. Data can be cut and pasted between any other Windows application or imported/exported through a spreadsheet like interface. Alternatively data may be exchanged using OLE.	Supervised and self-organising nets, MLP, Kohonen, RBF and NEUfuzzy. Skelotonisation and LM forecasting. Configurable network parameters. For novices NEUframe will automatically create an architecture based on the parameters of the training data. Implementation may be in NEUframe, from another Windows application such as Excel, using the NEUrun OLE runtime module, or by extracting C, C++, Java or Matlab code.	Limited by hardware only.	Recommended configuration – Windows 95 or NT, hardware spec to suit Windows and/or the application. Typical configuration: Win95 or NT, P120, 32-50 Mb disk space.	Ideal for developing solutions and embedding them in the Windows environment or onto hardware (e.g. process controllers, SMART cards, portable data terminals, etc). Easy to use for non-specialists. A mode menu leads first timers through the steps required to build a NN. Single click operation for most functions and sensible default parameters for novices with override facilities for more experienced users.	From Neural Computer Sciences ((http://www.ncs.co.uk). A free evaluation copy is available from the web-site. Educational discount available. Additional modules: • Radial Basis Algorithm • NUEfuzzy – world leading neuro-fuzzy developed with Southampton University. • NEUrun OLE runtime • Code Extract (C, C++, Java & MATLAB).
Neural Connection ™	Graphical interface, tool palette, tool bar, menu bar and spreadsheet input. One click access to descriptive statistics. Other tools include split datasets, filter tool, Box-Cox transformation, time-series window. Multiple linear regression, closest class means classifier, principal components analysis.	MLP with one or two layers and up to 4,000 nodes. RBF with centre optimisation, up to 65,535 centres and non-linear functions. Kohonen network with 1 or 2 dimensions and variable number of nodes. What if? Tool lets you explore your model with colour plots. NetAgent scripting language for interactive applications and batch jobs, and a INI scripting language for building applications	15,000 case maximum, 750 maximum inputs. General guideline for dataset size according to SPSS: use 10(M+N) training records; M predictions, N attributes. Handles missing data.	386 or better PC. Windows 3.1 or later, 4 Mb RAM (8 Mb recommended), 4 Mb free disk space, math co-processor strongly recommended.	Build models using logical icon-based tools which are easy to learn, move and connect with your mouse. Interactive applications and application building require learning the scripting languages, which will increase learning time. On-line help, users' guide and tutorial.	From SPSS (http://www.spss.com
NeuralWork Professional ™ II/Plus	A flexible development tool with an icon-based tool palette and mousable menu. Includes a variety of diagnostic tools and options for quick and easy building networks. Custom instruments can be created. Can automate development process using batch files written in C. Can be customised through a text-based scripting language	Over 20 different algorithms including common network types such as MLP, Kohonen and RBF. Each major class of network has its own custom menu with options specific to that class.	Limited by machine resources.	Sun workstation, INTEL 386, 486 and Pentium based PCs. Apple Macintosh, IBM RS/6000, NEC EWS-4800, HP 9000/700 and Silicon Graphics IRIS.	GUI identical for all versions. Requires programming. Broad range of functions. Requires a good knowledge of NNs. Not recommended for end-users. Technical support includes telephone or e-mail hot line under annual support agreement. Training available on-site or through courses. Over 1,000 pages of documentation includes a tutorial, an in-depth reference book on neural computing and a user's guide.	Developed by NeuralWare Inc. (http://www.neuralware.com) and supplied by Scientific Computers. (http://www.scl.com). Advanced, comprehensive package. Prices vary depending upon configuration and platform.

Product	Overview	Features	Data limits	Platform	Ease of use	Comments
Neuralyst for Excel™	A general purpose NN engine integrated with Excel. A self-programming network, that trains itself on the data and goals set, reporting on its progress. Excel charts can be used to measure NN characteristics, and its formulas used to pre- and post-process data.	Includes a genetic supervisor which can take a NN configuration developed by the user and optimise it for the best performance or fastest computation time. The NN engine is written in C and implements the backpropagation algorithm.	Excel spreadsheet limits: 256 columns and 6500 rows of data. The NN configures with 6 layers and supports over 13,000 neural connections. 500,000 connections per second after training or 150,000 connections updates per second on a 33 MHz 486 PC.	For IBM-PC or compatible: Windows 3.1 or higher, Windows 95 or NT. Macintosh system 7.0 or higher. Requires 1 Mb of disk space. It can be ported to other platforms if required.	Uses the interface of Excel and adds two new menus to the menu bar. Data is selected from the columns of the Excel spreadsheet and goals set using the spreadsheet language. Most tasks require a click of the mouse. The familiar interface makes Neuralyst easy to use.	A trademark of EPIC Systems Corporation and licensed to Cheshire Engineering Corporation. Demonstration available for download from (*http://websites.earthlink.net/ ~chesheng.index*)
NeuroGenetic Optimiser ™ (NGO)	A NN development tool that uses genetic algorithms to optimise the inputs and structure of a NN.	Algorithms include fast backpropagation, time delay, continuous adaptive time, probabilistic or generalised regression, selectable accuracy calculations, learning rate and momentum user setable, data splitting alternatives.	Maximum 512 inputs, 1 or 2 layers 1 hidden layer, 256 neurons, 100 outputs. Processor intensive with large networks running for days even on fast machines.	IBM compatible 386/486/Pentium PC with 8 Mb memory, Windows 3.1 or higher and 1.3 Mb free disk space for installation.	Setup Wizard to guide users, on-line context sensitive help with full manual contents.	From BioComp Systems, Inc. (*http://bio-comp.com*).
NeuroShell ™	NN development program with several algorithms. NeuroShell uses a worksheet data file format and can call up your own spreadsheet. Cannot generate C-code networks. Has internal graphics and a data viewer.	Choice of 15 architectures including Kohonen, three-layer back propagation, jump connection, probabilistic, general regression and some proprietary algorithms such as Netperfect which helps prevent overtraining.	No information.	Windows 3.1 or above IBM PC compatible computer with 80386 or higher processor, 4 Mb RAM and 5 Mb hard disk space. Customer support is unlimited and responsive.	Icon-driven Window-based package, includes advanced and beginner systems. Commercial users can access networks from C, Visual Basic or Excel. On-line help.	From Ward Systems Group Inc. (*http://www.wardsystems.com*).

5.4. References and Bibliography

Baptist R. P., C. E. Yeary, R. J. Slutz and V. E. Derr. 1994. Comparison of Neural Network and Linear Regression Models for Predicting El Niño Events, National Oceanic and Atmospheric Administration, Silver Spring, MD, Environmental Research Laboratories.

Bishop, C. J. 1995. Neural Networks for Pattern Recognition. Clarendon Press.

Buckles, B. P. and F. E. Petry. 1992. Genetic Algorithms. IEEE Computer Society Press.

Caudill, M. 1990. Neural Network Primer. Miller Freeman Publications.

Chauvin, Y. and D. Rumelhart. 1995. Backpropagation: Theory, architectures and applications. Lawrence Erlbaum Association.

Dabiri, A. E., M. Garrett, T. Kraft, J. Hilton and M. van Hammersveld. 1993. Application of Neural Networks to Waste Site Screening, EGG-WTD-10677, Department of Energy, Washington.

Dowla, F. U. and L. L. Rogers. 1995. Solving Problems in Environmental, Engineering and Geosciences with Artificial Neural Networks. MIT Press.

Hashem, S. 1995. Environmental and energy applications of neural networks. Workshop on environmental and energy applications of neural networks conference, Richland, WA., PNL-SA-25920, CONF-9503142-1, Department of Energy, Washington, DC.

Jensen, F. V. 1996. An Introduction to Bayesian Networks. UCL Press.

Kanevskij, M. F. 1995. Using artificial neural networks for the spatial interpolations of radioecological data. CODEN: IAETAV, 3, 26-33.

Kartalopoulos, S. V. 1995. Understanding Neural Networks and Fuzzy Logic : Basic Concepts and Applications. IEEE Press.

Kasabor, N. K. 1996. Foundations of Neural Networks, Fuzzy Systems and Knowledge Engineering. MIT Press.

Keller, P. E., R. T. Kouzes and L. J. Kangas. 1994. Three neural network based sensor systems for environmental monitoring. PNL-SA-24175 CONF-9405197-1, Battelle Pacific Northwest Labs., Richland, WA., AC06- 76RL01830; FG06-89ER75522, Department of Energy, Washington, DC.

Kirschfink, H. and R. Weber. 1993: Using fuzzy tools in traffic data analysis. Proceedings of the Second Meeting of the EURO Working Group on Urban Traffic and Transportation, 15-17 September, Paris, France, 349-356.

Kosko, B. 1992. Neural Networks and Fuzzy Systems. Prentice-Hall.

Michalewicz, Z. 1994. Genetic algorithms + data structures = Evolution Programs. Springer-Verlag.

Mitchell, T. M. 1997. Machine Learning. McGraw Hill.

Orlov, Y. V., I. G. Persiantsev, S. P. Rebrik and S. M. Babichenko. 1995. Application of neural networks to fluorescent diagnostics of organic pollution in water. In Proceedings of SPIE – Society of Photo-Optical Instrumentation Engineers, Bellingham, WA, USA, 150-156.

Radknight, C. M., G. R. Balls, G. E. Mills and D. Palmer-Brown. 1997. Modeling complex environmental data. IEEE Transactions on Neural Networks, 8, 852-861.

Rogers, L. L. 1992. Optimal Groundwater Remediation using Artificial Neural Networks and the Genetic

Algorithm. Ph.D. Thesis, Lawrence Livermore National Lab., CA; UCRL-LR-114125, Department of Energy, Washington, DC.

Schalkoff, R. J. 1997. Artificial Neural Networks. MacGraw Hill.

Schmuller, J. 1990. Neural networks and environmental applications. In: Expert Systems for Environmental Applications, edited by J. M. Hushon, American Chemical Society, Washington, DC, 53-68.

Smith, B. L. and M. J. Demetsky. 1995. Traffic Flow Forecasting for Intelligent Transportation Systems. VTRC-95- R24, Virginia Transportation Research Council, Charlottesville; FHWA/VA-95/R24, Federal Highway Administration, Richmond, VA., Virginia Div., Virginia Dept. of Transportation, Richmond.

Swingler, K. 1996. Applying Neural Networks : A Practical Guide. Academic Press.

Transportation Research Board. 1994. Intelligent Transportation Systems: Evaluation, Driver Behavior, and Artificial Intelligence. TRB/TRR-1453, Washington, DC.

Veelenturf, L. P. J. 1995. Analysis and Applications of Artificial Neural Networks. Prentice Hall.

Wagner, J. S., M. W. Trahan, W. E. Nelson, P. J .Hargis and G. C. Tisone. 1994. Chemical Recognition Software. SAND-94-2817C, Sandia National Labs., Albuquerque, NM; CONF-9411144-3, Department of Energy, Washington, DC, AC04-94AL85000, Optical sensing for environmental and process monitoring, McLean, VA.

Weiss, S. M. and C. A. Kulilowski. 1991. Computer Systems that Learn: Classification and Prediction Methods from Statistics, Neural Nets, Machine Learning, and Expert Systems. Morgan Kaufmann.

6. EXPERT SYSTEMS

6.1. Capabilities and Limitations

The goal of the chapter is to give you sufficient information to decide if expert system technology is appropriate for your problem. As with neural networks, many people find the term ‘expert system’ intimidating. This chapter should help you feel more comfortable with the concept, and give you the essential facts you need to get started.
Before you read further, consider the following:

1. Even with good supporting tools, expert systems are usually harder to create than systems based on neural or statistical approaches. This is because somebody must thoroughly understand all of the rules and the way they interact, and this requires skill and time. Nevertheless, there are situations where an expert system approach is the best one. The primary benefit is understanding: the behaviour of an expert system is always completely consistent with a set of facts and rules that you control. This is not to say that the behaviour will always be what you expect; as the number of rules gets larger, it becomes difficult for humans to hold them all in mind. But it will be consistent and, at least in principle, you will be able to work out how it arrived at each result.

2. In many business expert system communities, an often used payoff threshold is 160 Kecu. If the predicted savings or earnings multiplied by the success probability of the project is above 160 Kecu, then it is reasonable to consider using expert system technology. The implication for environmental analysts is clear: if your problem is technically suitable for an expert system approach, then check first that the benefit of the solution will be sufficiently significant to justify the costs; and evaluate the risk of failure carefully. New expert system technologies such as those we will introduce in this chapter should help reduce this payoff threshold, but it is too early to say by how much.

What are expert systems?
Expert systems are tools for reasoning with information that is expressed as facts and rules. They have two modes: one in which the system is given facts and rules and another in which it answers questions based on the facts and rules.

Fact: France is a country.

Fact: France is a member of the European Union.

Fact: In France the legal limit for pesticides in groundwater is 0.2 microgram/l.[15]

Fact: European law sets the legal limit for pesticides in groundwater as 0.1 microgram /l.

Rule: If (country) is a member of the European Union, then the environmental regulations of (country) take precedence over those of the European Union.

Question: What is the legal limit for pesticides in groundwater in France?

Answer: 0.2 microgram /l.

When there are many facts and rules, then the ability of an expert system to deliver correct responses can be very useful, because these systems can handle large tasks easier than humans. They will not forget, and they will not make mistakes. There again, they are only as good as the

[15] This is a hypothetical figure.

information they have been given; and the rules may contradict one another, although a good system should help sort this out.

One famous type of expert system is chess-playing programs, but these were developed by a world community of enthusiasts, over a long period of time, and they are only useful for this narrow task. Expert systems are like that: they require high intelligence and dedication to create, and once created, can serve only the narrow domain for which they have been designed.

Why are they important?
Expert systems capture the knowledge and understanding of experts about a problem domain, and make that understanding available for use when the experts are unavailable. Doctors, for example, can ask expert systems for assistance in diagnosing and treating difficult conditions. A wide variety of organisations are making use of expert systems. A survey of UK companies in 1991 found applications in fields as diverse as production, marketing, customer support, training, regulation compliance, and security. However as yet, little use has been made of expert systems for environmental management; although particularly in the domain of legislation, they seem to offer much potential. Another area in which they could prove useful is risk assessment. Here, the importance could be not so much in the automation of the reasoning process, as in its careful definition and documentation. A third area, in which there are several examples, is interpretation of remote sensing data[16].

How are expert systems different from neural networks and statistics?
Expert systems are important when you need something that not only produces useful numbers, but also includes understanding. This is a fundamental difference between expert systems and statistics or neural networks; interest in the ‘why’ of behaviour.

There is a strong analogy between expert systems and physical-based computer models. When we build physical models of systems, we gain the ability to make predictions, and the quality of those predictions is used to signal whether or not our understanding of the system (expressed as a set of equations) is correct. Expert systems are like that; except that rather than a set of physically based equations, one has a set of facts and rules. While physical models work on measurable parameters such as temperature and acidity, expert systems work on relationships and judgements such as guilt/innocence or stop/go. To create an expert system, a tool is used to describe some facts and rules, then the tool is used to ask questions and display answers.

What are the differences among the various types of expert systems?
There are three different kinds of expert systems:

- conventional expert systems: for black-and-white facts and rules
- fuzzy expert systems: for vague or imprecise knowledge
- Bayesian or belief networks: for probabilistic situations.

When it comes to comparing the various types, the different proponents have been very critical of other methods (see August 1995 issue of Technometrics, e.g. Almond, 1995, Bonisone, 1995, Kandel et el, 1995, Laviolette et al, 1995, Rousseeuw, 1995 and Zadeh, 1995). No one seems to have a problem with conventional expert systems, but many statisticians (who favour Bayesian networks) are vehemently against the fuzzy concept, and vice versa. In what follows, we will try to explain the differences and at the same time indicate the roots of the deep questions that cause such strong feelings.

Conventional expert systems
These are also known as ‘crisp’ AI systems. Conventional expert systems reason with symbols using binary logic. ‘If the switch is ON and the light is OFF, the bulb or the circuit may be at fault’.

[16] For example, the EU-funded ‘IceRoutes’ project, led by EOS, is applying expert system and neural network technology to the identification of ice types using space-based radar imagery. See http://www.eos.co.uk.

There is no in-between with a switch, nor can the light be anything else but on or off. The output of a conventional expert system is a crisp statement. Adding the fact ‘the bulb is OK’ allows the expert system to conclude ‘the circuit is faulty’. To allow the system to handle the possibility that the power had failed, additional rules are required.

Rule-based programming is one of the most commonly used techniques for developing crisp AI systems. In rule-based programming, rules specify a set of actions to be performed for a given situation. A rule is composed of an ‘if’ portion and a ‘then’ portion. The ‘if’ portion specifies the conditions that cause the rule to be applicable. The ‘then’ portion defines the action(s) to be performed. This is called a ‘production rule’. A rule based system acts when it is given a fact or facts. It begins by finding an applicable rule and carrying out the specified actions (which may add or remove facts). Then it finds another applicable rule and executes its actions. This process continues until no applicable rules remain, at which point it responds with a true or false.

The beauty of this concept is that the programmer does not need to worry about how the computer should perform its task (which is the usual job of a programmer). The rule-based programmer’s job is getting a consistent and complete set of facts and rules. Nevertheless, this is a task requiring considerable skill, because it takes mastery of both the expert system technology and the problem domain from which the facts and rules come.

At one time, many advocates of conventional expert systems felt that all of human knowledge could be captured in this way. The ongoing CYC project in the USA is attempting to give enough knowledge to an expert system to allow it to behave like a human in ordinary situations. CYC now incorporates around a million such rules, but is still not able to meet its objective. The reason is that humans unconsciously call on a vast body of information in making even the simplest decisions: ‘Should I take an umbrella? I see light glistening on the raindrops on the window, so I will.’ To teach a computer to reason thus is astoundingly complex. Things like ‘the curtains are closed’ result in long and clumsy chains of inference that are difficult to manage[17].

The lesson is that: within strictly limited domains, success is achievable, but wide domains invite failure. This rule applies to all types of expert systems.

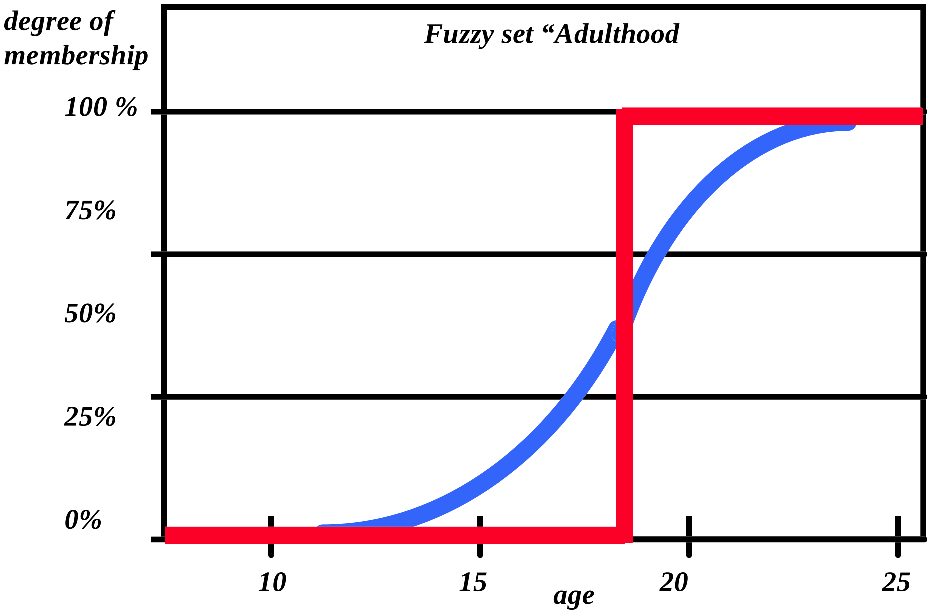
Fuzzy expert systems (FES)

The idea here is that in many situations, things are not black and white at all, rather there are vaguely defined aspects that require a different way of reasoning. For example: ‘if the battery is normal and the light is dim, then the bulb may need replacing or the contacts may be dirty’. Fuzzy systems offer a simple arithmetic for reasoning with such information that is effective in a wide variety of practical situations. There are far more deployed fuzzy systems in the world than there are conventional expert systems or Bayesian networks. Some of these are quite impressive. For example the Tokyo subway system, which has no drivers, just a fuzzy system with a small number of rules. Even more impressively, a voice-controlled fuzzy helicopter piloting system (Mockler, 1992).

The central idea in fuzzy logic is the fuzzy set. In normal set theory an element is either a member of a set or it is not, i.e. its ‘membership function’ is 1 or 0, and there is no possibility of it being a partial member. In contrast fuzzy set theory offers the notion of being a partial member of a set, by allowing elements to have membership values that lie between 0 and 1. Figure 6.1 contrasts the two approaches. The concept of adulthood is represented as a set based on age. A conventional set has everyone over 18 a member of the class adult and anyone below this value not adult. Fuzzy sets permit a gradual transition from not-adult to adult via the membership function.

[17] The heart of the difficulty is extracting the relevant concepts from the great many that could be considered. This is a deep problem which is far from being solved.

Figure 6.1: Comparison of fuzzy and normal sets



A fuzzy rule is a statement like:

IF (waste is hazardous) THEN (use safety precautions for removal)

Fuzziness allows you to deal with continuous degrees of hazard, and the corresponding variable degree of the need for safety precautions. Fuzzy expert systems are usually agreed to be more expressive than crisp expert systems. This has led to a reported 80% success rate in developing fuzzy systems as opposed to a 5% success rate with conventional knowledge based systems (Harris-Jones, 1995). A closely related technology, which we will explore under this heading, is ‘neurofuzzy systems’. These provide the capability to construct and tune expert systems based on example data, much like neural networks.

Bayesian networks

Bayesian networks (also called belief networks) are a statistically rigorous way of reasoning with probabilistic networks. If the battery is normal (probability 100%), and the light is dim (85%), then the chance that the bulb needs replacing is 60% and the chance that the contacts are dirty is 40%. Bayesian networks require you to be able to define probabilities and relationships among probabilities that completely cover the problem domain. [18]Given this information, they compute the probability of a particular decision being correct. A domain in which Bayesian network technology is particularly relevant is medical diagnosis. Here case histories and a large body of statistics should make it possible (at least in principle) to gather all the necessary information. However, we will defer giving you more information on Bayesian networks until later in the chapter.

What to consider in choosing an expert system?

Table 6.1 summarises the primary characteristic of each type and indicates the consequences for expert system development.

[18] There are other statistically based approaches that relax this assumption (Dempster Schaeffer, for example), but we have not found commercial products supporting them.

Table 6.1: Major types of expert systems

Type of system	Primary characteristic	Consequences
Conventional expert systems	Rules are black and white.	OK for appropriate problems, but clumsy when shades of grey arise.
Fuzzy expert systems	Rules come into effect gradually.	Good for control problems (hotter, colder) and other problems with a continuous output character.
Bayesian networks	Rules compute probabilities.	Requires a complete set of probabilities connecting nodes; these can be difficult to define in some situations.

The type of expert system tool you should select depends on what you need and what you have available. So before going further, ask yourself these questions:

Want do I need –

- facts – descriptions of the existing situation;
- predictions – statements and numbers about what could happen in the future;
- deductions – analysis of facts leading to useful conclusions;
- understanding – insights into why a situation is so.

What do I have available –

- data – examples of the phenomenon;
- expert knowledge – people who understand what is going on in detail;
- or both.

Table 6.2 sketches the different strategies options.

Table 6.2: Needs and strategies

Situation **Problem**	**Have data**	**Have expertise**	**Have both**	**Examples**
Need facts	Use statistics.	Ask the expert.	Evaluate statistics with expert.	Find evidence about X.
Need predictions	Statistics, neural.	Select from conventional, fuzzy, or Bayesian expert system.	Neurofuzzy or Bayesian / learning.	Given X, what will happen in future, or what would happen if X occurred?
Need deductions	Neurofuzzy or Bayesian / learning.	Select from conventional, fuzzy, or Bayesian expert system.	Neurofuzzy or Bayesian / learning.	What laws are applicable to X?
Need understanding or diagnosis	Fuzzy / neurofuzzy.	Select from conventional, fuzzy, or Bayesian expert system.	Neurofuzzy or Bayesian / learning.	What is the cause of X?

6.1.1. Key Capabilities

The key capabilities an expert system must provide are:

- learning – an efficient way to enter facts and rules
- query – a way to put questions to the system and get answers
- explanation – the ability to explain how a response was reached.

Learning
The process of creating an expert system normally takes at least two people: an expert and a knowledge engineer. The knowledge engineer gets the expert to explain his/her reasoning, and then puts the facts and rules into the expert system. Maintaining consistency among the facts and rules is important, and this can be rather difficult if the number of facts and rules is large (even one hundred is surprisingly challenging). The different types of expert system have radically different ways of expressing rules and facts, as we will discuss below.

Query
The ways the different kinds of expert systems are queried differ considerably, but the query interface is usually not the most important factor in deciding what type of expert system you need.

Explanation
The ability to explain how a conclusion was reached is of great importance, because this is how expert systems are tested. Before the knowledge engineer unleashes the system on the world, the expert must test the system by asking it as many different questions as possible. When an answer seems anomalous, the expert asks it to explain how it came to the conclusion. This uncovers contradictory or incomplete information, so corrections can be made.

6.1.2. Limitations and Likely Evolution – the Authors' View

The major problem that expert systems have is that they do not scale well – that is they handle small problems well, but the additional effort required to handle larger problems grows too rapidly. The core problem is 'how do you know that a large expert system is correct?' For a small system, you can test it by asking it all the possible questions. But the number of possible questions grows very rapidly as the number of facts and rules increases. Beyond a certain point, one runs out of testing budget, and is left with a complex system, the correctness of which cannot be determined. There are various strategies for overcoming this barrier.[19]

For crisp AI systems, one popular strategy is black-board based architectures. Black-board based systems try to solve problems by committee. The idea is to assemble a collection of small expert systems, each capable of handling some part of a larger problem. They all sit around watching a 'blackboard' on which questions and facts appear. When one of them sees a bit of the problem that it can handle, it does so, putting its results on the blackboard. The intelligence of the group is obviously greater than that of the individual experts. However, blackboard-based expert systems have been criticised as performing slowly and as not easy to maintain. In general, it may be said that the rate of progress in crisp AI systems is not as rapid as with the other two types.

Turning to fuzzy systems, the neurofuzzy approach seems to hold much promise for the future. The strategy is to integrate the strengths of fuzzy systems, neural networks and statistical reasoning. In particular, the latest developments are in the direction of the 'ideal' statistical performance offered by Bayesian networks. Current fuzzy / neurofuzzy technologies make it easier to develop expert systems, but they do not overcome the size barrier. Future developments will result in improved learning algorithms and better scalability. Bayesian networks are also rapidly evolving in capability, and the parallels with neurofuzzy systems are quite strong. Recent Bayesian network products can determine probabilities from data, and work under development will further extend their capability.

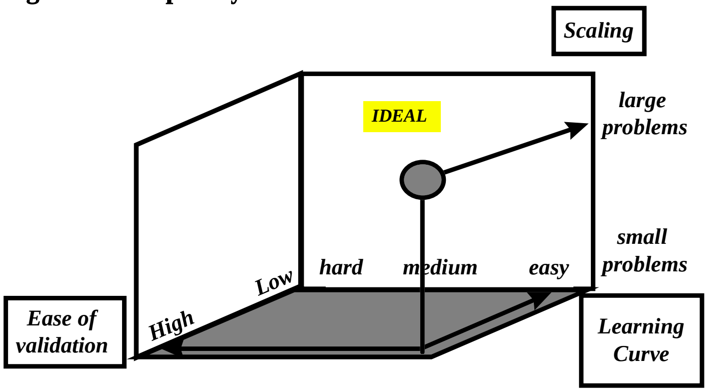
We see the most likely future as one in which fuzzy/neurofuzzy and Bayesian approaches merge. Crisp AI systems, we believe, will be a suitable solution only for very restricted classes of problems.

[19] What do we mean by large? There are expert systems that deal with hundreds or thousands of inputs, but in these systems each input has a small number of states – for example the on/off states of switches. When the inputs have a large number of states or are continuous, the 'curse of dimensionality' limits the number of inputs sooner.

6.2. Representative Expert System Tools

Figure 6.2 shows three of the factors that are most important in selecting an appropriate expert system tool.

Figure 6.2: Expert system selection factors



An 'ideal' tool would be suitable for both large and small problems – its evaluation time would not become unacceptably long as the problem got larger.[20] It would have an easy learning curve, and would not be difficult to validate. Two other important factors are ease of extension and cost/availability. The first of these requires explanation: because expert systems are rather expensive to develop, if they are successful at all, they tend to have long lifetimes. Over this long lifetime, new requirements are identified, and a need develops to expand the boundaries of the original system. Some technologies are very difficult to extend, while others are more malleable. This is related to, but distinct from, scaling, which is can it do the calculations, while extensibility is can you do the programming.

It is also important to consider the types of inference the tool supports, i.e. prediction / diagnosis / learning / optimising / planning; and whether it works well with discrete variables, continuous variables or both. In this section we will look at one from each of the main types of expert system followed, in Section 6.2.4, with a listing of other major systems currently on the market.

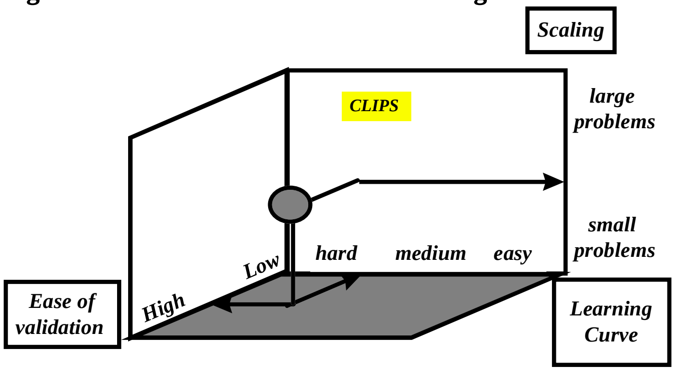
6.2.1. CLIPS™ – for constructing Crisp AI systems

CLIPS (C Language Integrated Production System) is a crisp AI expert system shell used extensively by NASA and the US military. The primary goals of CLIPS are to provide portability, efficiency and functionality. CLIPS provides a development environment for constructing rule- and/or object-based expert systems. It also has procedural programming capabilities, and a fuzzy extension (which we do not discuss here). Our ratings on CLIPS are shown in Figure 6.3 and discussed below.

Scaling
The time it takes for a rule-based expert system to give an answer depends on the number of rules that need to be evaluated. This in turn depends on how well structured the rules and facts are, which is a function of programming skill and complexity of the problem domain. As the evaluation of a rule is simpler than for fuzzy or probabilistic networks (it is either true or false), rule-based systems can be expected to perform more quickly. On the other hand, much of the work that a rule-based system does is evaluating which rules are applicable; a pattern-matching task that the others do not (at present) attempt. Of the three types of systems, it is hardest to predict the scaling behaviour of crisp AI systems.

[20] So you need to project how your ambitions will evolve – an inappropriate product will disappoint you if your challenge grows rapidly, but if your expectations are unlikely to grow, purchasing more power than you need to get started may be a waste. Also, increasing machine power can make up for some lack of scalability.

Figure 6.3: CLIPS™ selection factor ratings



Learning curve

CLIPS is a tool for programmers, and ought not to be approached by the uncommitted. Training in rule-based programming is a prerequisite. The standard version of CLIPS features an interactive development environment that includes on-line help. The primary interface is a text editor, although CLIPS does provide some windows to view the current state of the knowledge base and dialogue boxes to examine/manipulate the knowledge base. CLIPS is fully documented and includes a Reference Manual and User's Guide. Support is available by e-mail. The CLIPS version 6.0 Users' Guide (May 1993) is written in a humorous style that gradually communicates the CLIPS concepts. Considerable other CLIPS documentation is available on the WWW.

Ease of validation

Validation is the hardest job with any expert system and CLIPS is no exception. The difficulty is magnified over fuzzy expert systems and the Bayesian network that we review here by the lack of a graphical user interface.

Ease of extension

Crisp AI systems tend to have tree-like structures: general rules and facts lead to more specific ones. This causes a serious difficulty when a major extension to the tree is required – many or all of the existing facts and rules may need to be modified to take into account the widened domain.[21]

Cost and availability

Copies of CLIPS can be obtained for free from web sites such as:

http://www.ghgcorp.com/clips?WhereCopy
http://www.jsc.nasa.gov/~clips/CLIPS

CLIPS runs on Windows 3.1, Windows 95 and NT, MS-DOS and most UNIX platforms. Table 6.3 shows the cost factor rankings for CLIPS, using the categories defined in Table 2.1.

Table 6.3: Cost factor rankings for CLIPS™

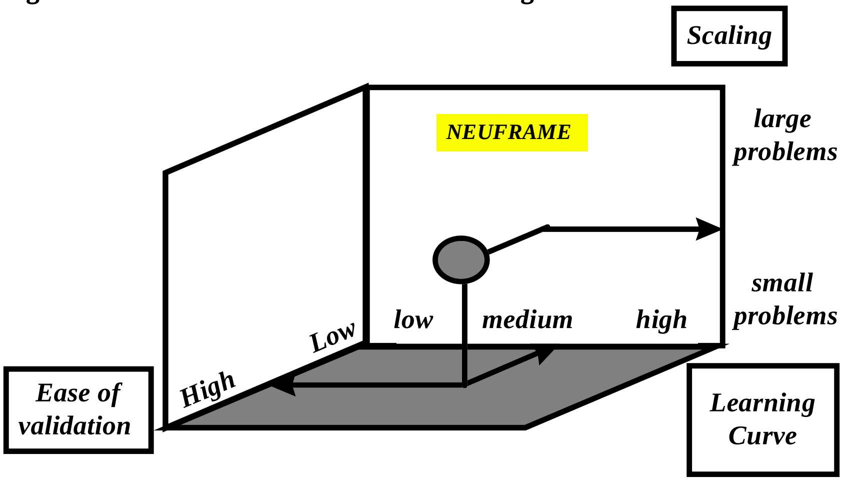
learning	installation	application	price
D weeks or months	B an hour of a specialist	D several weeks or months	free

[21] This problem was encountered by the authors when an attempt was made to extend an advisor expert system on the applications of ERS-1 spaceborne synthetic aperture radar data. We wanted to extend it to handle other satellites. Initially, it seemed that generalising it could be done in a few hours. But it was soon realised that all of the existing rules assumed that there was only one satellite. The predicted cost of the extending the system to cope with other satellites was so high the attempt was abandoned.

6.2.2. NEUframe™ – a fuzzy expert system tool

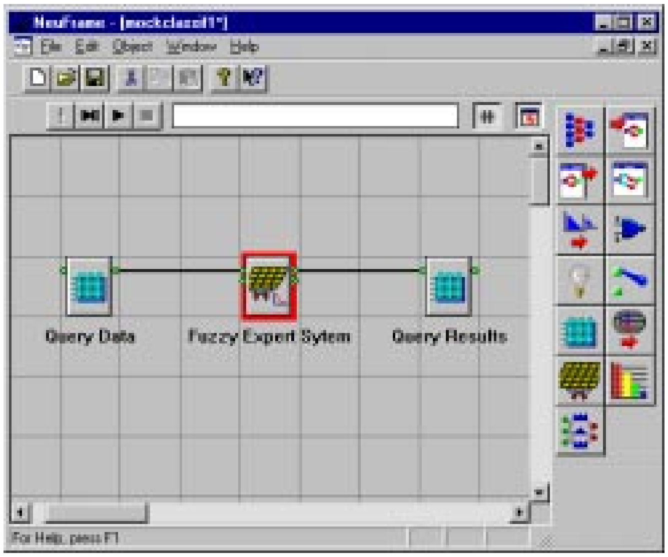
NEUframe from Neural Computer Sciences *(http://www.ncs.co.uk)*, is an example of a product offering fuzzy expert system capabilities. NEUframe also offers neural network and neurofuzzy features; but in this section, we focus on the fuzzy expert system aspects. Its ratings are shown in Figure 6.4.

Figure 6.4: NeuFrame™ selection factor ratings



The workspace is the main layout area of NEUframe where objects can be dragged, connected and moved as desired. The interface is carefully laid out, and the system is quite robust. Care has been invested to make features convenient, but sometimes they are so clever that it takes a while to find them. Queries and query results, indeed all kinds of data within Neuframe are handled by the datasheet (symbolised by the left and right icons in Figure 6.5).
The authors tested the datasheet with up to 50,000 rows x 10 columns of numerical atmospheric data without problems.

Figure 6.5: Neuframe™ main window

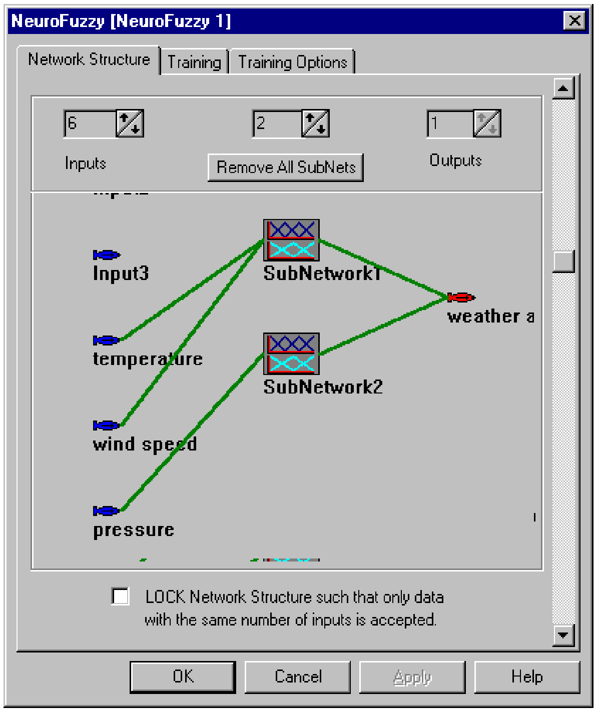


To create a fuzzy system, one identifies a set of sub-networks that capture the elements of reasoning in the problem, graphically connects the appropriate variables to sub-networks, defines fuzzy sets and fuzzy rules within each sub-network, and assigns rule confidences. Figure 6.6 shows various inputs from the data sheet being connected to sub-networks. Figure 6.7 shows three fuzzy sets that have been defined for a weather variable: pressure. The sets are labelled ‘falling’, ‘steady’, and ‘rising’. In Figure 5.8 simple fuzzy rules are seen corresponding to each. The rules are created and edited in an easy point-and-click dialogue within this window.

Scaling

Although there is no general rule for predicting performance, it is very easy with Neuframe to construct a fuzzy expert system with the number and types of connections you expect to need in your future systems. You do not need to fill in the actual rules and confidences to test out the evaluation time. We found simple fuzzy processing of 50,000 rows x 10 columns of data to be reasonably quick – substantially less than one minute.

Figure 6.6: Connecting inputs, sub-networks and outputs



NeuroFuzzy [NeuroFuzzy 1]

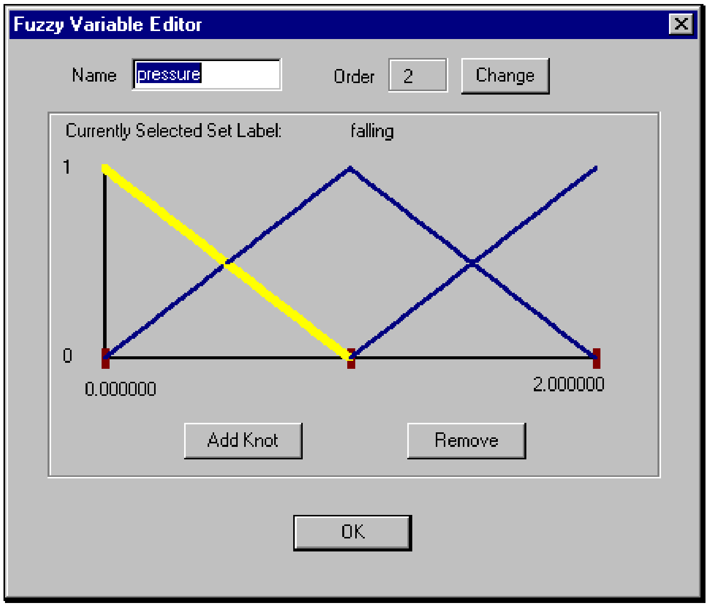
Inputs: 6

2

Outputs: 1

□ LOCK Network Structure such that only data with the same number of inputs is accepted.

Figure 6.7: Three fuzzy sets describe an input variable



Fuzzy Variable Editor

Name: pressure

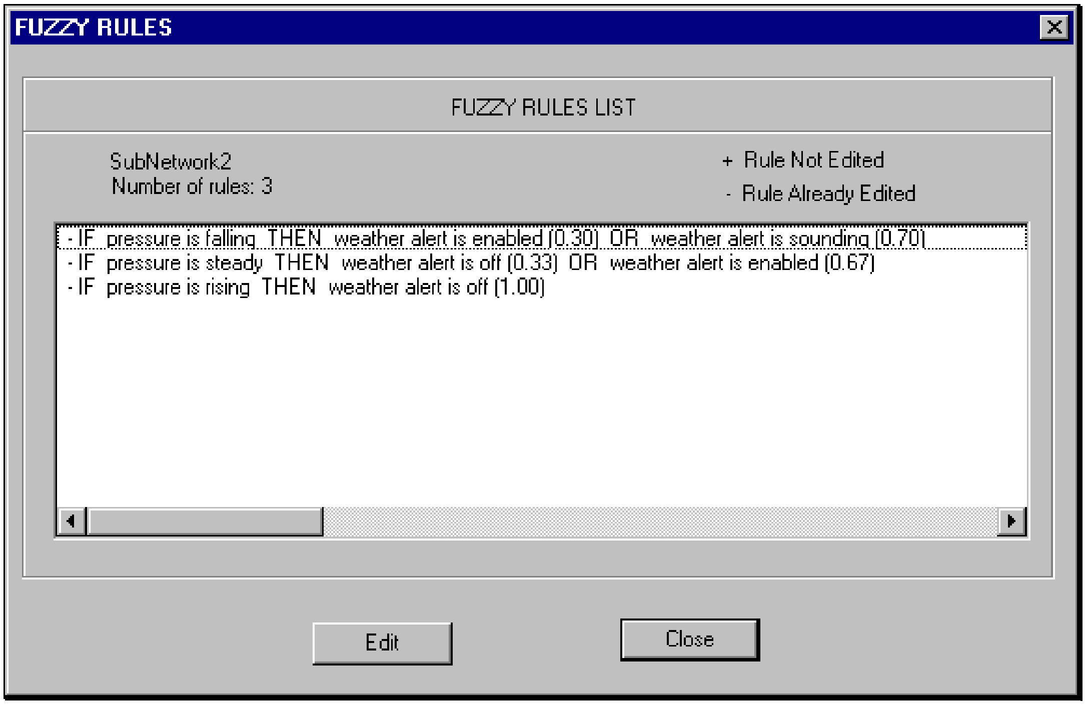
Order: 2

Currently Selected Set Label: falling

Learning curve

You should expect to invest a week or two familiarising yourself with the tools and the concepts before you will be ready to tackle your problem. Even then you should begin small and expect to iterate towards a satisfactory solution. A nice thing about Neuframe is that because it does not take much of your time on the mechanics, it is easy to do just this. NEUframe is provided with a brief printed manual and extensive on-line documentation. Useful tutorials that introduce various aspects of the tool in a step-by-step fashion are very helpful in getting started. NEUframe also offers cue cards and context-sensitive help. But although the on-line documentation covers all the main points, it does have gaps that leave the user puzzled at times. The lack of either a complete printed reference manual or a way to systematically print the on-line help may be found annoying by some.

Figure 6.8: A simple set of fuzzy rules



FUZZY RULES

FUZZY RULES LIST

SubNetwork2
Number of rules: 3

+ Rule Not Edited
- Rule Already Edited

- IF pressure is falling THEN weather alert is enabled (0.30) OR weather alert is sounding (0.70)
- IF pressure is steady THEN weather alert is off (0.33) OR weather alert is enabled (0.67)
- IF pressure is rising THEN weather alert is off (1.00)

Ease of validation

Validation is the hardest job with any expert system, and NEUframe is no exception. Two useful features of Neuframe that help the validation task are the ability to plot variables and see which rules have fired for each query. Another is the ability to process all the data in a table in one pass. This means that many validation examples can be tested quickly. Finally, extremely rapid development of fuzzy expert systems is possible with NeuFrame's neurofuzzy mode, provided that an appropriate variety of data examples are available; the system can either completely configure itself, or simply adjust the fuzzy weights and confidences. Automatically configured fuzzy systems tend to be less easy to understand than ones that are constructed by experts; thus this mode loses some of the point of having an expert system. However, for validation purposes, comparing the performance of a hand-crafted system against an automatically configured one can be very useful.

Ease of extension

NEUframe offers the capability to encapsulate units of reasoning inside 'components'. Components can hold fuzzy reasoning units, but also data sheets, neural networks, logic gates, and various kinds of encoders. This means that, in principle, more complex fuzzy expert systems can be constructed from simpler ones.

Cost and availability NEUframe is available from Neural Computing Sciences of Southampton UK (*http://www.ncs.co.uk*). A separate module 'Neurun' allows you to export NEUframe applications for embedding in Windows applications and another module provides the facility to export C, C++, or Java code. NEUframe runs on Pentium or 486 PCs under Windows, Windows NT or Windows 95. The recommended configuration is Windows NT 486DX 50 MHz or above with 16 Mb memory, 150 Mb or above hard disc, and VGA graphics. Table 6.4 shows the cost factor rankings for NEUframe using the categories defined in Table 2.1.

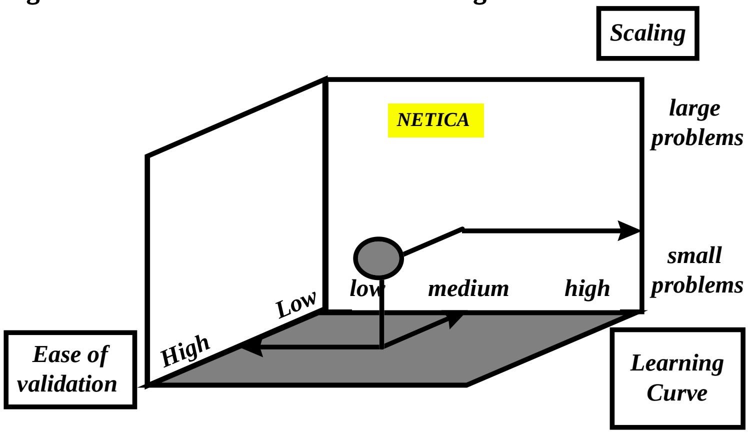
Table 6.4: Cost factor rankings for Neuframe™

learning	installation	application	price
C-D one to several days	A less than 10 minutes	C-D days to weeks or months	D+ includes neural network software

6.2.3. Netica™ – a Bayesian network tool

As we explained in Section 5.1, Bayesian networks capture believed relations between the variables relevant to a problem. The variables might be relevant because they can be observed, because we need to know their value to take some action, or because they help us express the relationships between the rest of the variables. The relations may be deterministic or imprecise (probabilistic). Our ratings for Netica are shown in Figure 6.9, and will be explained shortly. First, we need to explain the concepts in a little more detail.

Figure 6.9: Netica™ selection factor ratings

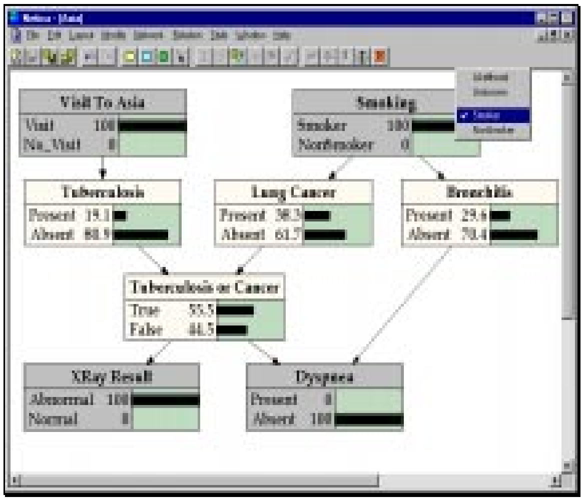


The basic ideas behind Bayesian networks are:

- the probability of something being true (say a patient having a particular problem with his lungs) depends on the evidence that is available (smoking or non-smoking, etc.)
- an expert can define probabilities connecting different states of evidence: P(A|B) – the probability of A given the fact of B
- a network program can automatically work out a set of consistent probabilities, based on evidence given by the user.

Netica is a PC-based Bayesian Network tool you can use to create probabilistic expert systems. It is also useful for finding patterns in data, and optimising decisions, although we will not discuss these aspects here. The kinds of applications Netica is suitable for include diagnosis, prediction, decision analysis, sensor fusion, expert system building, reliability analysis, probabilistic modelling, and data mining [22] . Figure 6.10 shows a Netica network for diagnosing lung conditions, greatly simplified for ease of illustration.[23]

Figure 6.10: A simple network for reasoning about disease



The greyed boxes are ones for which evidence has been found: the patient has an abnormal x-ray, no dyspnoea (trouble breathing), is a smoker, and has visited Asia (where he might have contracted tuberculosis). The other boxes show the resulting probabilities for and against each of the possible conditions. A diagnostician could work efficiently with such a tool by collecting more evidence to narrow down the options. Of course, a larger network could include all such evidence.

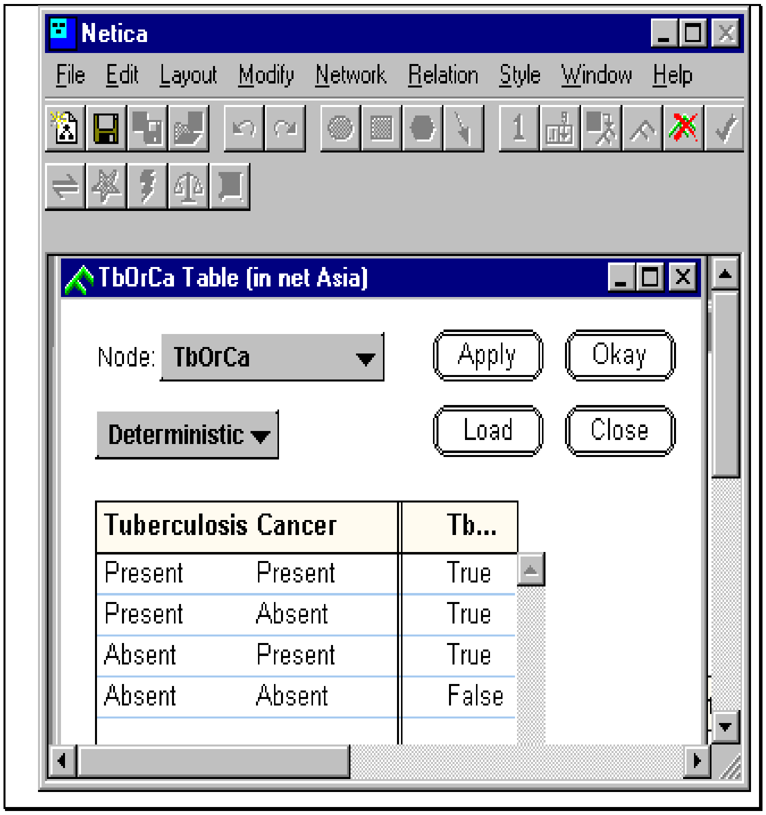
Netica offers dialogue windows such as those in Figure 6.10 and Figure 6.11 for forming and updating the relationships among variables. In Figure 6.10, a diagnostic network is shown being queried. A doctor is entering an observation relevant to one of the nodes: the patient is a smoker. This affects all of the other probabilities, which are immediately updated accordingly. In Figure 6.11, the set of probabilities governing the relations between nodes is being defined. Normally, Netica assumes that there is a set of discrete states that can be related, but it also accepts relations in the form of equations.

Scaling

The time it takes for Netica to update all the probabilities after you give it data depends on the size of the network and the number of interconnections. For a 37-node network it took much less than one second on a standard PC. It is difficult to make a general statement about what would happen with a larger network, for example 10 times larger in number of nodes. Roughly speaking, if the number of connections per node does not increase, then the evaluation time should increase linearly (10 times as long). But if the number of connections on nodes goes up sharply, performance will degrade more rapidly. Although there is no general rule for predicting performance, it is very easy with Netica to construct a network with the types of connections you expect to need in your future systems. You do not need to fill in the actual values for the probabilities to test out the evaluation time, because it will be almost constant regardless of what the values are or what query is made.

[22] Much of the above text is drawn from the Netica Users Guide, version 1.03 (with permission).
[23] One of the Netica examples is a realistic network for monitoring heart patients, consisting of 37 nodes.

Figure 6.11: Probabilistic rules


Netica
File Edit Layout Modify Network Relation Style Window Help
TbOrCa Table (in net Asia)
Apply
Okay
Load
Close

Node: TbOrCa

Deterministic

Tuberculosis	Cancer	Tb...
Present	Present	True
Present	Absent	True
Absent	Present	True
Absent	Absent	False

Learning curve

As with the other types of expert systems, you should expect to invest a few days or perhaps a week or two familiarising yourself with the tools and the concepts before you will be ready to tackle your problem. Even then you should begin small and expect to iterate towards a satisfactory solution. A nice thing about Netica (like Neuframe) is that because it does not take much of your time on the mechanics, it is easy to do just this. Netica is not as supportive as NEUframe. There is no on-line help, at least in version 1.04, that is reviewed here. The manual has some good aspects, but left considerable room for improvement.

Ease of validation

Validation is the hardest job with any expert system, and Netica is no exception. The ease with which facts can be entered makes this job somewhat easier than with Neuframe, but the fact that each must be entered manually counteracts this advantage. Netica offers several other features useful for validation, including:

- case files – the ability to store and retrieve the information entered about one or more actual examples
- random case generation – useful for generating test data
- parameter learning – working out probability tables from examples.

A deficiency connected with parameter learning in Netica is that it assumes independence between each of the conditional probabilities connecting a node and its parents.

Ease of extension

Netica has a number of features to ease the difficult task of extending the capability of your expert system. For example, it is possible to cut and paste whole sections of existing networks into others; and it is possible to collapse sections of a network, once it has been validated. These reduce, but do not remove the problem generic to all expert systems, that they become increasingly difficult to understand (and thus maintain and extend) as they grow in size.

Cost and availability
Netica is available from Norsys Software Corporation, Vancouver, Canada *(http://www.norsys.com)*. Versions for Windows 95, NT, Macintosh and UNIX are available. Table 6.5 shows the cost factor rankings for Netica using the categories defined in Table 2.1.

Table 6.5: Cost factor rankings for Netica™

learning	installation	application	price
C-D days to weeks	A few minutes	C-D days to weeks	A+-D depending on platform / options

6.2.4. Other Products

Table 6.7 provides summary details of some of the other expert system packages that are on the market today.

6.3. Environmental Applications

While expert system development began in the late 1960s, their utilisation in the environmental field only been much more recent and in many cases is still limited to process-oriented applications, such as the management of water or waste treatment plants. This slow emergence of expert systems relates primarily to the multi-disciplinary and complex nature of environmental issues, where few standardised analysis methodologies exist. Hushon (1990) in an overview of environmental expert systems suggested that, at the time, their application was hardware limited and any future development would require their transfer from PC-based systems to dedicated workstations and mini-computers. Now the pendulum has swung back; with major advancements in PCs and a growing selection of software packages the application of expert systems is available to a wide audience of users. Most of the vendors provide excellent materials at their web-sites on expert systems in general and their product capabilities, but specific details on applications in the environmental arena are often lacking. For example on the Lumina web-site *(http://www.lumina.com/ software/learnaboutanalytica.html)* it states that Analytica is being used for acid rain mitigation strategies and for monitoring global climate change due to the 'greenhouse effect', but no further information is provided.

Thus, our examples of applications later in this section concentrate primarily on two products – from this it should not be mis-interpreted that we recommend these products above all others

To counter this focus, Bonisone (1995) offers the following advice on tool selection:

- **Generality**: tools vary widely in how many types of problems they can address and how many different ways they let you control them – more general tools take longer to learn and are more difficult to use, so don't buy more generality than you need.
- **Problem type**: select a type of tool that matches your problem (conventional, fuzzy, or probabilistic).
- **Interface**: select a tool with built-in capabilities for explanation and a good graphical interface – this should reduce your development time.
- **Evaluation**: test the tool early by building a small prototype – if possible, find a way to multiply up data and rules so you have an indication how it will perform on the final problem.
- **Reference sites**: ask the vendor to supply you with names of people who have successfully built systems similar to the one you intend to build. Speak with these people; they are usually more

than willing to give you a few minutes, and what they have to say could be crucial to your decision.

In relation to the problem type Table 6.6 lists some ways that expert systems can be applied, and indicates which type is most suitable for each.

Table 6.6: Expert system application types

	Conventional expert systems	**Fuzzy system**	**Bayesian network s**	**Comments**
Training	Yes.	Possibly.	Possibly.	Usually, people learn best from precise examples. Later they develop ways to deal with vagueness and probability.
Data management	Yes.	No.	No.	The flow of information in a data management system is a binary process.
Regulation	Yes.	Yes.	No.	Although regulation is meant to be crisp, there are often shades of meaning and contradictory rules that must be considered.
Diagnosis	Possible, but not the best approach.	Possibly.	Yes.	Inferring the most likely cause of an environmental event requires careful reasoning with data that is continuous and often subject to uncertainty.
Prediction	Possibly for restricted kinds of prediction tasks.	Yes.	Possibly.	In the absence of enough data to train a neural network, prediction is most easily done with fuzzy systems. Probabilistic networks have many more parameters that can be difficult to determine.

In addition to finding out more about expert systems from other sources (see also Section 6.4), there are some useful independent web-sites, such as IEEE Expert (*http://www.computer.org/pubs/expert/1995/features/* x60012/gei.htm) and periodicals such as ‘Expert Systems’, which has a particularly useful newsletter section on applications, tools, techniques and meetings.

The following paragraphs cover a small selection of case studies to illustrate examples of the use of expert systems in:

- agriculture
- forestry and agroforestry
- traffic management
- waste water treatment
- air quality monitoring.

Agriculture

Davidson and Williams in the Agricultural Research Service of the U.S. Dept. of Agriculture, *(http://www.exsysinfo.com/Appnotes/peanut.html)* have used the MS-DOS version of EXSYS to develop an expert system to help manage irrigated peanut production (EXNUT). EXNUT compiles data from an individual peanut field throughout the growing season and makes recommendations for irrigation, the application of fungicides and pest management. It optimises irrigation management based upon information about the peanut plant, soil, weather, insects and plant diseases. The scientists felt an expert system provided the best way to deliver technology to the farmers and the EXSYS system shell was chosen due to its ease of use, its ability to run external programs and access external data files. The ability to examine the reasons for each decision was also a feature farmers desired. EXNUT has been evaluated on over 50 farms in Georgia and has consistently produced higher yields and quality, using less water and fungicides, than those managed by ‘expert’ farmers. Versions have been developed for the different conditions and peanut varieties grown elsewhere in the USA, while other expert systems are in development, which will make decisions on variety selection, land preparation and harvest scheduling.

Dryland salinisation is considered the worst land degradation problem facing farmers in Southern Australia today. The problem of salt in the soil can be treated by the proper management of groundwater, but until recently there has been no attempt to develop a long-term management strategy for dealing with the groundwater fluctuation that causes excess salinity. In response the University of Adelaide has developed GISintegration, a system integrating an expert system built with LispWorks and KnowledgeWorks with a GIS built using ARC/Info and Oracle *(http://www.harlequin.com/products/ads/lispworks/aus-farmers.html).* The GIS provides maps of an area so that farmers may access them via the expert system. The farmers enter information regarding various parameters affecting salinity on their land, such as soil or elevation, into the expert system, which imports it into the Oracle RDBMS. GISintegration can then interrogate the data and analyse it based upon production rules. Another expert system developed with KnowledgeWorks is called DRY-PLAN, which aids agricultural officials in determining what type of crops suit the soil of a particular dryland area. DRY-PLAN is meant to be more portable than GISintegration and will eventually run on a portable PC used during consultations in the field.

Forestry and agroforestry

Forest inventory is defined as a mathematical procedure for obtaining information on the quality and quantity of standing forest timber. Traditionally, forest inventories are conducted by expert foresters who are familiar with the forest topography and timber quality. An expert system in forest inventory is appropriate since the best sampling method and its cost depends on several quantitative, qualitative and other intangible factors, whose influence can best be expressed as a set of if-then rules. The University of Georgia's School of Forestry has developed a system using EXSYS *(http://www.exsysinfo.com/Appnotes/forest.html)* that considers two most widely used forest inventory sampling procedures – line plot and point sampling, each of which can be stratified. The system recommends an appropriate sampling procedure, the plot size to be used for sampling and estimates the cost of the inventory procedure. The major criterion for selecting the sampling method is the estimated relative cost of the procedure as a function of the desired level of accuracy, and the forest topography. EXSYS Professional was also selected as the tool *(http://www.exsysinfo.com/Appnotes/agroforestry.html)* to develop the United Nations University Agroforestry Expert System (UNU-AES) to assist land-use (agricultural, forestry, etc.) officials, research scientists, farmers and individuals interested in maximising benefits gained from applying agroforestry approaches to land management for sustainable production of food and fuelwood supplies by farmers in developing countries. EXSYS was selected as the tools as the software was required to run under MS-DOS and to operate on a portable computer. In addition to being simple to learn and use the tool's important features included a screen definition language (to customise the way questions are asked), its ability to allow interface calls to external programs and an explanation facility to assist the user. Inputs to the system included annual rainfall, number of rainy days/yr, elevation, slope, soil texture, soil fertility, and soil reaction data. Knowledge pertaining to socio-economic characteristics can also be built into the system, as well as other agroforestry technologies such as fallows, plantation crop combinations, home gardens, soil conservation hedges, etc..

Traffic management

To ease traffic management in 34 km of tunnels and covered highways around Neuchatel, in Switzerland, the French engineering firm Cegelec *(http://www.gensym.com/customerstories/cegelec.htm)* have developed an operator decision support system using G2. For traffic guidance alone, operators face the daunting task of managing more than 900 traffic lights and 650 Variable Message Signs (VMS), with over 28,000 data points controlling these signs. "We needed an operator decision support system that would enable operators to guide traffic around incidents safely and efficiently by automatically generating and executing control scenarios in the case of emergencies" reported Daniel Demode, project leader at Cegelec. With the expert system, they were able to capture the knowledge and guidelines of the Neuchatel Police and built the system in just 10 months. The system reduces the decisions to be made by the operator to a few mouse-clicks. When an operator receives information about an incident along the highway (from video input, emergency response teams, or road crews), he or she highlights on a map of the network which highway segments need to be closed off. The system then calculates the control strategy to redirect traffic around the closed-off segments from its knowledge base of traffic flow and behaviour, safety procedures, geographic constraints, and operating procedures. Finally, upon confirmation by the operator, the expert system executes the control strategy by sending signals to the traffic lights and VMS' along the highway. G2's real-time data processing capabilities allow it to simultaneously monitor and control thousands of instruments.

In another traffic management system Huang et al (1994) used an expert system to optimise traffic flow during busy periods, identify stalled vehicles and accidents, and aid the decision-making of an autonomous vehicle controller.

Wastewater treatment

When AVEBE *(http://www.gensym.com/customerstories/vertis.html)*, the world's largest producer of potato starch, began construction of a new wastewater treatment plant, it chose to develop an intelligent operator decision support system using G2 software, which was selected because it provides a powerful rule-based environment and a flexible graphical user interface. "With G2's object-oriented architecture and easy-to-use graphical user interface, we completed prototypes in record time – perhaps 10 times faster than if we had been using C++" reported software engineer Maarten Wetterauw. The initial objective was to enable inexperienced weekend production staff to control the plant effectively when regular operators were absent. In time, the system would play a key role in helping the entire plant comply with tough environmental regulations. By correlating the problems most frequently encountered with the corrective measures taken the decision support system prompts the operator to answer questions before reaching a conclusion and offers the operator advice on corrective measures. Before putting the advice into practice, operators can simulate the effect on the plant using dedicated simulation software. Even at an early stage of the AVEBE project, positive results are evident. The weekend operators have much better control over the plant, and the decision support system has been an effective training tool. Most importantly, the system releases the company's environmental engineers from time-consuming troubleshooting at the beginning of each week. With the short term goal accomplished, the aim is now on reducing operating costs by lowering the plant's nitrogen discharges. By examining the relationship between different production processes, wastewater quality and plant functioning, adjustments can be made to stop common problems from re-occurring. Wetterauw explains that "the Dutch government levies a fee per kilogram of nitrogen discharged into the sewer system. G2 will help us lower costs by monitoring and controlling that discharge more effectively".

In the UK, Yorkshire Water Services *(http://www.gensym.com/customerstories/yorkshire.html)* selected G2 to address both business and process issues at their 150 treatment plants. Their water works vary in size – processing between 10 cubic metres and 270 megalitres per day. When they were faced by a manpower reduction, they looked to an expert system to take over some of the responsibility. They wanted to link monitoring, diagnostic, and problem-solving logic across a network to prevent excursions from the norm. G2 was selected for its graphical, object-oriented interface, being easy to use, simple to create new schematics, and the ability to write rules in a natural, almost English language. In the past each filter was washed every 24 hours, regardless of its condition – using rules and models, the wash sequence was modified to improve its performance. In the future Yorkshire Water plans to use Gensym's Telewindows to link their G2 solution across multiple sites and replace a variety of older SCADA systems. They will also use Gensym's NeurOn-Line to help with their pH control system, where problems tend to be non-linear and vary over time.

Air quality monitoring

The increasing number of environmental regulations and subsequent added complexity of power plant control systems have caused increased demands on control room personnel. ABB Power Plant Control (Mannheim, Germany) and PreussenElektra have developed an expert system for monitoring and fault analysis of power plant processes which will provide operator decision support. The ABB 'Optimax 9000 Expert' (model-based diagnosis) product *(http://www.gensym.com/customerstories/abbpower.html)*, built with G2, is intended to detect problems in the operation of fossil-fuelled power plants. With the aid of the expert system, control room personnel can detect any discrepancies from the optimum process status and more readily analyse the cause. They can therefore plan and take corrective actions more effectively and more quickly. Diagnoses are performed in accordance with fault tree diagrams, which associate root-cause problems with symptoms reported by instrumentation, and in accordance with power plant models. When the cause of a fault is determined, this information is displayed to the operator along with a recommended response.

In California the South Coast Air Quality Management District (SCAQMD) is a regional government agency that was formed to create and enforce air quality regulations for four counties. As the air pollution control agency for metropolitan Los Angeles, SCAQMD is battling the worst air pollution in the United States, and has the strictest pollution control requirements worldwide. In 1991, SCAQMD's board passed Rule 1135 which limited and required monitoring of the nitrogen oxide (NOx) emissions from five electric utilities in the LA basin. To monitor this Rule, it is stipulated that the utilities measure boiler emissions at set intervals each day and transmit the data in near real time to SCAQMD. The agency's challenge was to develop a system that could determine rule compliance on a daily basis. Because of the rule's complexity, the large volumes of data, the real-time factor, and the desire to limit operator interface, they could not use traditional methods of data collection and analysis. Instead, SCAQMD chose G2 *(http://www.gensym.com/customerstories/scaqmd.htm)* to create compliance objects, relationships, attributes, and rules that represented the intent of the legislation. SCAQMD's G2 application enables them to monitor plants in real time from portable computers communicating over mobile phone lines and replaces the reams of strip charts they used to receive. The time and personnel savings to the utilities companies has been estimated at over $750 million.

.

Table 6.7: Other expert systems products

Product	Overview	Features	Data limits	Platform	Ease of use	Comments
Analytica™	Visual environment for creating, analysing and communicating probabilistic models for risk and decision analysis. The successor to Lumina's Demos decision modelling system. Influence diagrams.	Hierarchical influence diagrams provide a visual display of the model structure. Uncertainty about any variable is expressed by selecting a probability distribution. Uncertainties propagate through the model using Latin hypercube or Monte Carlo sampling or display uncertain results as standard statistics, probability bands, probability density functions, or cumulative probability functions. Developers can build and distribute applications which access the Analytica Decision Engine on Windows 95 and Windows NT platforms.	14,900 variables, 15 dimensions per array, 32,000 elements per dimension, 32,000 sample size.	Macintosh 68020 and later; 2.5 Mb RAM for 68000 version; 4 Mb RAM for Power Macintosh version, System 6.0 and up. Ported to the PC platform in 1997.	Intuitive GUI. Uses a small number of standard diagram symbols making it easy to use and learn. Buttons and nodes are simply clicked to evaluate a model and nodes positioned and arrows drawn between them to create a graphical model as an influence diagram.	From Lumina Decision Systems (*http://www.lumina.com*). Demonstration are available for viewing or downloading.
Eclipse™	Production system with a syntax similar to CLIPS. Supports object-oriented programming with C++, automatic integration with standard databases. Optional case-based reasoning integration.	Uses the Rete algorithm to support forward and backward chaining and is independent of the number of rules. The Easy Reasoner is a Case-based reasoning product which learns how to access stored information and reasoning capabilities that adapt prior experience to new situations. It recalls previous problems with similar symptoms and uses them to solve current problems. It uses statistical techniques to automatically discover the conceptual structure within records stored in databases.	Scales to thousands of rules. Faster and smaller than CLIPS.	Windows 3.1 with Win32s, Windows 95 and NT.		From Haley Enterprise Inc (*http://www.haley.com*). Versions available for download.
EXSYS Professional ™	EXSYS provides a multi-level approach to expert system development by providing a suite of tools, combining many advanced features.	Knowledge-based development tool that features a rule editor and compiler, command language, automatic validation, backward and forward chaining, blackboarding, linear programming, SQL interface, frames, report generator, fuzzy logic, neural net, tree diagramming, security, 6 confidence modes. Embeds and interfaces to other applications, databases and process control software. Applications are completely portable across platforms.		Versions available for PC: Windows 3.1, 95, NT, Macintosh and UNIX: Sun, HP, SGI, VAX.:	Developed to meet the requirements of complex expert systems. Aims to balance the conflicting requirements of flexible power and ease-of-use. To do this, EXSYS provides four levels of tools with increasing capability allowing developers to use the easiest tool that will meet their needs. The tools are designed so that you need only learn a small portion of the commands to build most expert systems.	From EXSYS Inc (*http://www.exsysinfo.com*). Free demonstrations available and an Application Advisor helps you determine which tools to use for a specific expert system project and ranks the difficulty of the project and tells you where most of the development work will be.

Product	Overview	Features	Data limits	Platform	Ease of use	Comments
G2™	Application development environment for building and deploying intelligent applications. Handles real-time problems efficiently. Its tools include Bayes-On-Line (BOL) and the G2 Diagnostic Assistant (GDA).	BOL allows application developers to work at the component level, specifying the most immediate causes and effects of an event. These relationships can be used to develop end-user graphics. Can model behaviour over time for dynamic phenomena. GDA allows development and deployment of real-time applications for monitoring, diagnosis, decision support, quality management and intelligent supervisory control. Provides system for predicting and identifying process problems and assisting operators in determining the correct course of action. Its block components include statistical process control, decision trees, operating rules, procedure enforcement, fuzzy control and smart alarming.	More than one hundred pre-defined graphical blocks.	Windows 3.1 on Intel-based PCs, Windows NT on both Intel and DEC, platforms. UNIX platforms.	The GUI development toolkit (GUIDE) permits the development of a Windows-like interface. Users can organise and configure components to build real-time applications on-line. The graphical environment eliminates the need for programming, making the software easier to use for non-specialists.	From Gensym (*http://www.gensym.com*). The vendors provide a consultancy service which includes application engineering, surveys, prototypes, project management, knowledge base reviews and specialised training.
HUGIN™ System	A graphical expert system shell for construction and execution of Bayesian belief networks. It consists of five main components: an inference engine, an application program interface (API), a compiler, a run-time system and an editor.	The knowledge base is represented by a Bayesian belief network or influence diagram. The API allows the inference engine to be used from programs written in C. Using the API, the inference engine functions as a program library. The compiler structures the network for use by the inference engine. The run-time system displays the model graphically in a window environment from where users can add additional evidence on the state of the nodes. The editor creates and maintains the belief networks by creating nodes and links. There is a facility for compressing sparse probability tables or approximating tables to free up space for faster processing with little effect on the end results.	Allows construction of Bayesian belief networks of any size, limited only by the amount of virtual memory available.	PC 386DX or higher processor (Windows 95 / NT) or Sun-Sparc station. or	The graphical interface makes the system easy to use. The Net language for writing the belief networks will take a little longer to learn.	From Hugin Expert A/S (*http://www.hugin.dk*). Tutorials for the construction of a Bayesian belief network and the construction of a small influence diagram are available from the web site.
Knowledge Works™	An integrated development environment for constructing complex knowledge-based systems..	Enables developers to combine rule-based, object-oriented, logical, functional and database programming using Common Lisp, CLOS, OPS, Prolog and SQL Object-oriented knowledge representation, forward chainer based on Rete algorithm and backward chainer is an extended Prolog based on Warren Abstract Machine. Programming intensive.	Limited only by hardware memory.	UNIX workstations including: DEC Alpha, OSF/1; DEC MIPS, Ultrix; Sun SPARC, SunOS; Sun SPARC, Solaris; SGI MIPS, IRIX; IBM RS6000, AIX; HP PA 9000/700, HP-UX.	Requires a lot of programming and would therefore not be suitable for most end-users. Developer would require expertise in expert systems.	From Harlequin (*http://www.harlequin.com*).

Product	Overview	Features	Data limits	Platform	Ease of use	Comments
Strategist™	A graphical Bayesian network and influence diagram tool. Users can graphically model and analyse the decision situation including both influence diagrams, for showing the structure of the relationships between variables, and multi-dimensional arrays, for capturing the exact form of relationships between variables.	Classical Bayesian belief networks, hierarchical influence diagrams, multi-dimensional arrays, a language for representing the relationship between variables, continuous variables, a test studio for running the model through real-world situations, comprehensive set of analysis functions, optimal policy computation, value of information, sensitivity analyses, arbitrary marginal, joint or conditional query capability, a separable multi-platform inference engine for building embedded decision support application. It integrates expert judgement and data mining results through classical theories and modern technologies for modelling for decision-making.		Windows 3.1 (with Win32s), Windows 95 and NT. IBM compatible 486 PC or better, 8 Mb RAM and 10 Mb free disk space.	The visual environment makes Strategist easy to use and learn. Influence diagrams are easy to build and do not require prior knowledge of a programming language.	From Prevision Inc (http://www.prevision.com). Available under an 'early adapter' program which includes a free upgrade to the full release version.
XpertRule ™ KBS	Tool for knowledge acquisition and system development. Features inductive rule generation, genetic algorithms for optimisation tasks and a tool for building graphical user interfaces.	Includes a graphical development environment and XpertGen a C source code generator allowing integration of XpertRule solutions on most hardware/software platforms. Automatic rule induction generates efficient decision tree. Structured Decision Tasks (SDT) helps through the stages of problem modelling, knowledge structuring and knowledge acquisition. Advanced features include a command language, graphical dialogue editor to create custom GUI presentations, connectivity to hypertext help facilities, ODBC, DDE and DDL support to link the application to other programs and data sources and OLE2 automation.		Windows 3.1 or later, Windows 95 and NT, 80386 or higher processor, 8 Mb RAM.		From Attar Software (http://www.attar.com). An evaluation pack is available from the web site.

6.4. References and Bibliography

Almond, R. G. 1995. Discussion: fuzzy logic: better science? Or better engineering? Technometrics, 37, 267.

Baldwin J. F., T. P. Martin and B. W. Pilsworth. 1995. Fuzzy and Evidential Reasoning in Artificial Intelligence, Research Studies Press.

Bishop, C. S. 1990. Introduction to Expert Systems, Addison-Wesley.

Bonisone, P. P. 1995. Discussion: fuzzy logic control technology: a personal perspective. Technometrics, 37, 262.

Brown, M. and C. Harris. 1994. Neurofuzzy Adaptive Modelling and Control. Prentice Hall.

Davis, J. R. 1996. Expert systems and environmental modelling. In: Modelling Change in Environmental Systems, edited by A. J. Jakeman. M. B. Beck and M. J. McAleer. John Wiley & Sons.

Harris-Jones, C. 1995. Knowledge Based Systems Methods: A Practitioners Guide. Prentice Hall.

Heckerman, D. 1995. A Tutorial on Learning with Bayesian Networks. Technical Report MSR-TR-95-06, Microsoft Research. (see: *http://www.afit.af.mil/Schools/EN/ENG/LABS/AI/BayesianNetworks/*).

Heckerman, D., A. Mamdani and M. P. Wellman. 1995. Real-world applications of Bayesian networks. Communications of the ACM 38, 24-26.

Huang, T., D. Koller, J. Malik, G. Ogasawara, B. Rao, S. Russell and J. Weber. 1994. Automatic symbolic traffic scene analysis using belief networks. Proceedings of National Conference on Artificial Intelligence, Morgan Kaufmann.

Hushon, J. M. (ed). 1990. Expert Systems for Environmental Applications. American Chemical Society.

Jackson, P. 1990. Introduction to Expert Systems, Addison-Wesley.

Jensen, F. 1996. Introduction to Bayesian Networks. Springer Verlag.

Jensen, F. V. 1996. An Introduction to Bayesian Networks, UCL Press.

Kandel, A, A. Martins and R. Pacheco. 1995. Discussion: on the very real distinction between fuzzy and statistical methods. Technometrics, 37, 276.

Kosko, B. 1992. Neural Networks and Fuzzy Systems. Prentice Hall.

Laviolotte, M., J. W. Seaman, J. D. Barrett and W. H. Woodall. 1995. A probabilistic and statistical view of fuzzy methods. Technometrics, 37, 249.

Mockler, R. J. 1992. Developing Knowledge Based Systems using an Expert Systems Shell. Prentice Hall.

Rousseeuw, P. J. 1995. Discussion: fuzzy clustering at the intersection. Technometrics, 37, 283.

Russel, S. and P. Norvig. 1995. Artificial Intelligence: A Modern Approach. Prentice Hall.

Spiegelhalter, D. J., A. P. Dawid, S. L. Lauritzen and R. G. Cowell. 1993. Bayesian analysis in expert systems. Statistical Science, 8, 219-283.

Sugeno M. 1995. Intelligent control of an unmanned helicopter based on fuzzy logic. Proc American Helicopter Society, Texas, 791-803. (See also short reports: *http://www-cia.mty.itesm.mx/isai/sugeno.html* and *http://schooner.cs.arizona.edu:5605/japan/kahaner.reports/helicopt.92*)

Terano T., K. Asai and M. Sugeno. 1994. Applied Fuzzy Systems. Academic Press.

Zadeh, L. A. 1995. Discussion: probability theory and fuzzy logic are complementary rather than competitive. Technometrics, 37, 271.

7. OPTIMISATION AND RISK MANAGEMENT

7.1. Capabilities and Limitations

The goal of this chapter is to give you sufficient information to decide if there are optimisation and / or risk management technologies that are appropriate for your work. We will refer to these collectively as 'ORM'. ORM technologies, which have until recently been applied only by the most advanced organisations, are now widely available. They are not too difficult to use, and can help you shift from managing your domain in an ad-hoc fashion to a more efficient quantitative basis. For example, optimisation can help minimise the costs of operations. Risk management can help ensure that you achieve what you set out to do on projects, and can also be used to analyse environmental hazards such as floods.

At the time of writing, it was estimated[24] that less than 1% of European establishments were taking advantage of the significant benefits of these technologies. This is surprising, because unlike some other CIPTs, ORMs have been around for a long time. However, the rate of take-up is said to be increasing.

How do these technologies differ from the others in this manual?
All the other CIPTs have been about capturing information: finding out what the numbers say and do not say, or capturing human expertise in a way that it can be used by computers. The CIPTs featured in this chapter assume that you have done all this and what you need is help in taking an important decision, such as, for example, the best set of operating parameters for a waste treatment plant, or planning a project. The most important idea in optimisation is linear programming, while in risk management, it is Monte-Carlo methods. Other buzzwords you will encounter in this domain are genetic algorithms and non-linear optimisation. Here are some definitions:

Linear programming (LP): a well-understood collection of matrix-based techniques for optimising first-order (linear) quantities with relationships among the variables that are linear equalities or inequalities. LP is a widely-used business technology.
Non-linear optimisation: a less complete and robust collection of techniques that apply when the relationships are non-linear.
Monte-Carlo methods: a simulation technique useful in a wide variety of problems where the complexity of the problem makes analytical solution impractical or too costly. It builds up situation statistics by choosing random values for variables, simulating an outcome, and repeating this many times.
Genetic algorithms (GAs): a general problem-solving technique based on ideas borrowed from evolutionary theory. GAs are a special kind of Monte-Carlo method.

On the surface, linear programming and Monte-Carlo methods are very different concepts, and indeed optimisation and risk management are topics that could each be treated in a separate chapter. But because the kinds of problems these CIPTs can be applied to are closely related, they have been gathered here.

What are the different optimisation technologies?
Linear programming is the technology that is used to solve the great majority of the world's optimisation problems. It is essentially a clever way of handling sets of linear equations. The idea is that you define a cost function – something that you want to maximise or minimise, as a linear function of a number of variables. You also identify equations that must be satisfied among the variables, either equalities or inequalities; these are called 'constraints'. When the problem is set

[24] Grenville Croll, Eastern Software Publishing, private communication.

up in this way, LP tools can very efficiently find a guaranteed optimal solution. The reason LP is so popular is not so much the power of the method (there are other more powerful tools) but rather how easy it is to set up the problem. This is because many problems are inherently linear, and even non-linear problems often have useful linear approximations, and these are very straightforward for people to understand. The theory of LP is reasonably accessible, since many of the basic concepts can be visualised easily. See for example (Jakeman et al, 1993).

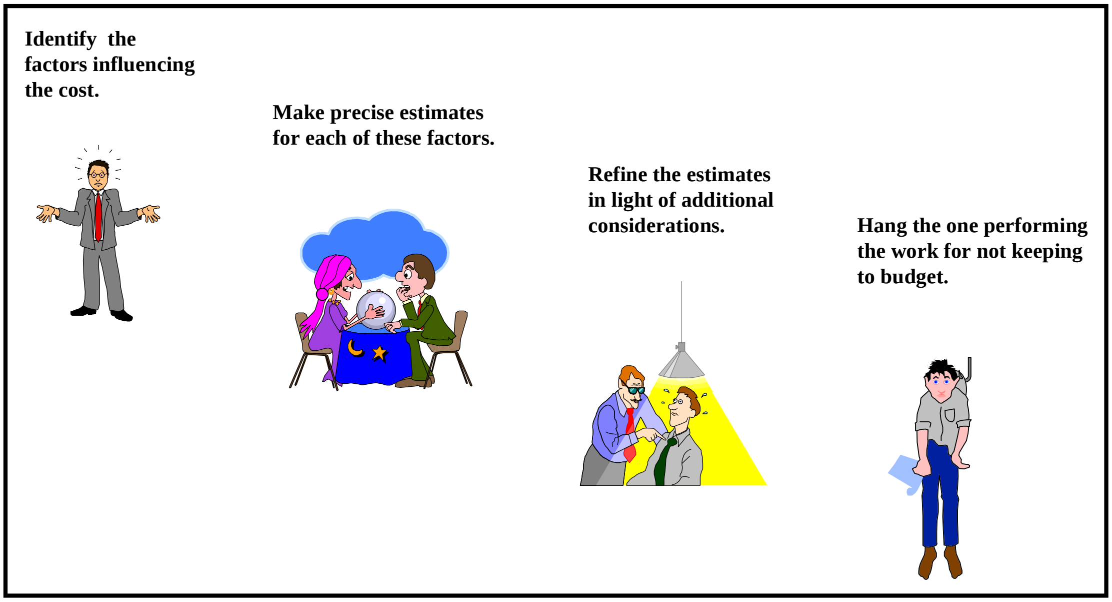
Non-linear optimisation tools can extend the linear programming idea to allow your equations to be highly non-linear, including higher order equations, trigonometric functions, and logic functions. The disadvantage is they involve additional complexity and loss of robustness.

An even more powerful approach, in principle, is **genetic algorithms**. Genetic algorithms are a special kind of **Monte-Carlo method**, which incorporates ideas from evolution. The idea is to set up a population of candidate solutions, select the most successful of them (those which get a better value of the cost function) and allow these to breed and mutate. Again the most successful are selected, and the process repeats thousands of times. Over the generations, the population evolves towards an optimal solution. The attraction of genetic algorithms is that there are no assumptions at all about the form of the equations. However, they are extremely inefficient compared to conventional methods, and more importantly it is difficult to guarantee robustness. So unless your problem has some highly non-linear aspect that rules out using linear or non-linear programming, genetic algorithms are best ignored.

What do risk management tools offer?

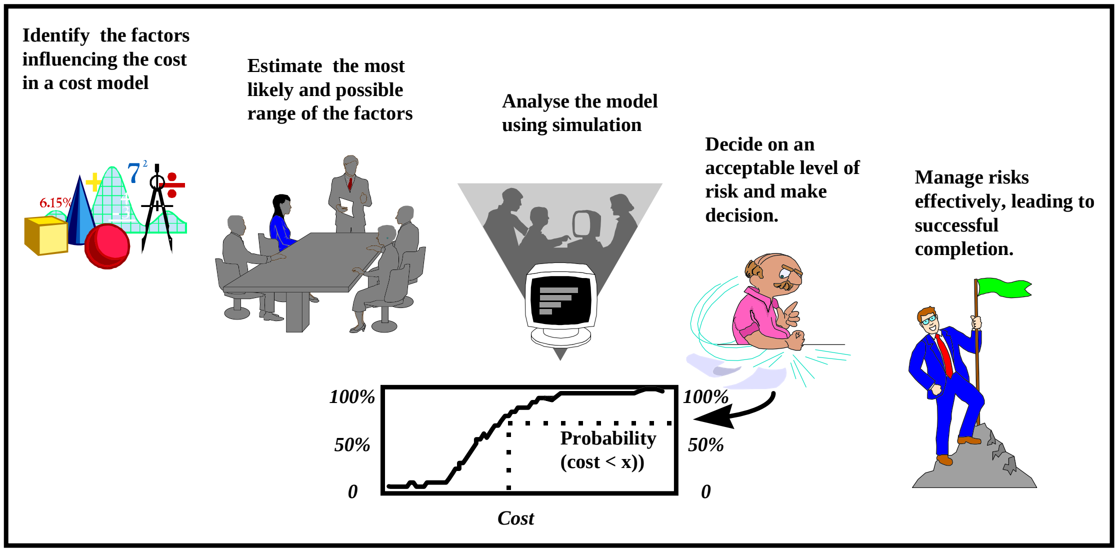
The typical process used to plan and control work is illustrated in Figure 7.1.

Figure 7.1: Typical practice in planning and controlling work


Identify the factors influencing the cost.
Make precise estimates for each of these factors.
Refine the estimates in light of additional considerations.
Hang the one performing the work for not keeping to budget.

There is now a better way, as illustrated in Figure 7.2.

Figure 7.2: A risk management approach


Identify the factors influencing the cost in a cost model
6.15%
7
Estimate the most likely and possible range of the factors
Analyse the model using simulation
Decide on an acceptable level of risk and make decision.
Manage risks effectively, leading to successful completion.
100%
50%
0
Probability (cost < x))
Cost
100%
50%
0

The short answer is better control of risk, especially cost and schedule risk, but also other kinds of risk. The new generation of risk management tools offer a major advance in power, without significant increase in complexity. In Section 7.2.3 we review a product that supports the point of Figure 7.2. Of course, the final frame still could be as in Figure 7.1, no guarantees are offered!

The reason managing things is so difficult is that it's about predicting and controlling the future. Everything is fixed in the past, but nothing is fixed in the future. This means that any prediction is uncertain to some degree. For example, a project may involve many different tasks. The network of tasks may be complicated, and the duration and cost of each task may be difficult to estimate precisely. A single overrunning task could seriously disrupt the schedule and increase costs, and multiple overruns are not uncommon. Managers must commit to budgets and schedules before all of the relevant facts are known, and do so under significant pressure, such as in negotiations with customers. In the past there has been little to help the manager know what the likely impact of a concession will be, given all the other uncertainties.

This situation has now changed. Risk management CIPTs can help predict, and thus control, the influence of uncertainties in task estimates. For someone considering the use of risk management technology, there are just two basic concepts to learn: the probability distribution and the principle of Monte-Carlo simulation. Many readers will have encountered these concepts; but for those who have not, here is a brief tutorial.

If something varies, then the question 'how does it vary' arises. Time of day varies uniformly: all times are equally likely. Weather varies non-uniformly: rain is more or less likely, depending on season and location, for example. A probability distribution answers the question 'how does it vary' with a graph showing how often each condition occurs. Figure 7.3 shows a possible rainfall probability distribution.

The graph says that the most likely daily rainfall is around 5 mm, and that rainfall greater than 10 mm is rather unlikely. Distributions can be integrated to produce 'cumulative distributions', as shown in Figure 7.4. The probability that the daily rainfall will be less than 5 mm is 45%. The cumulative distribution is the form used in most cases. A cumulative probability distribution is a good way to express uncertain information because it is quantitative, and because it is intuitively clear what it means.

Figure 7.3: Rainfall distribution

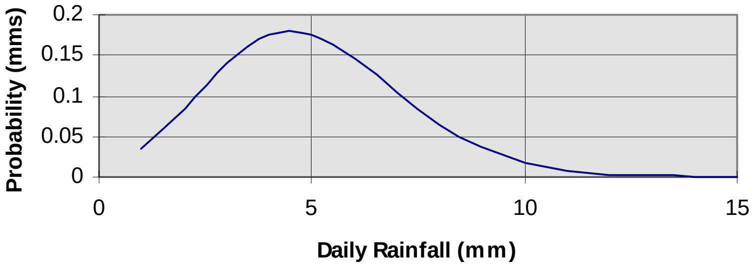
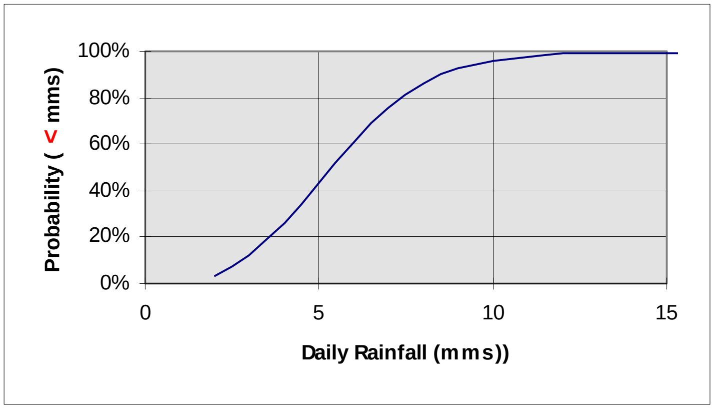
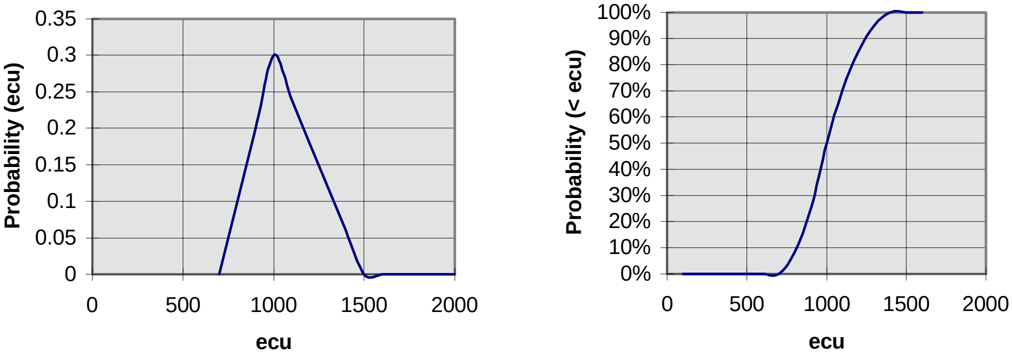


Figure 7.4: Cumulative rainfall distribution



The idea of the cumulative distribution can be applied to any kind of distribution. If you believe that the cost of doing a piece of work is most likely to be 1000 ecu, but it could be as low as 800 or as high as 1500, then you have some important knowledge about the situation that should be used. If you need to consider carefully the risks and consequences, it is much better to capture your intuition about the situation in a distribution than to pick a single number. Figure 7.5 shows this example.

Figure 7.5: Estimating distribution functions from experience



Suppose you have a number of factors that together determine the outcome in which you are interested; for example costs of components, labour rates, requirements expansion or government policy. For each of these you can find or define a distribution function. Monte-Carlo simulation samples randomly from all of these cumulative probability distributions to produce one example of what could happen. It does this over and over again, building up statistics on the possible outcomes, using chance to obtain a representative sample of the many different possibilities that could occur for each factor. Risk management tools use Monte-Carlo simulation, then produce convenient statistics on the outcomes. This is useful for two reasons:

1. It helps you inform yourself about the net risk implied by your model, and
2. It helps you understand the influence of the various factors that make up that risk.

7.1.1. Key Capabilities

The sections above have introduced the key capabilities needed for optimisation and risk management. We repeat them here to summarise.

Optimisation: maximising a function
When there is a quantity that needs to be maximised (or minimised), and there are conditions that set limits on what is possible, and there is a way to express the relationship between this and the variables that control it, then optimisation technologies can find the set of control variables that gives the absolute maximum (minimum).

Risk Management: integrating possibilities
When there are many possibilities, each uncertain to some degree, what one needs to know is the net risk and the contribution to the net risk of each component. With this information, you can concentrate on the major risk components. This capability is particularly applicable to project management. Conventional budgeting and project planning do not take into account the fact that all estimates have a degree of uncertainty. It can be shown mathematically that if there are concurrent *tasks, uncertainties will cause the critical path estimate of the cost/time to completion to be optimistic, typically by 15%.*

7.1.2. Limitations and Likely Evolution – the Authors' View

Optimisation technologies will become key factors in reducing the damage to the environment caused by man's activities. The lack of appropriate data gathering systems is currently the main restraining force, but this barrier will fall rapidly with the growth of mobile communications technologies. We suggest that although the capabilities of non-linear solvers will grow steadily, linear programming will continue to be the workhorse for a long time to come, because they are so easy to understand. Genetic algorithms will eventually come to play an important role, because they can solve all classes of optimisation problem, and are even easier to apply than linear programming. This will only happen when computing power is so abundant and cheap that the inefficiency of genetic algorithms is acceptable.
In general, the scale of the problems that can be attacked with optimisation technologies will grow, in both directions. Easy to use tools will mean that it is cost-effective to optimise smaller-scale processes. On the top end, we can envision the Dobris report[25] evolving into a quantitative tool in which optimisation tools are used to identify measures to achieve European-scale or even global policy objectives. However, it must be realised that the optimisation tools require numerical factors. Thus final decisions must be made taking into account intangible factors such as public opinion, political will and compromise. Of course, these can also be quantified, if somewhat subjectively.

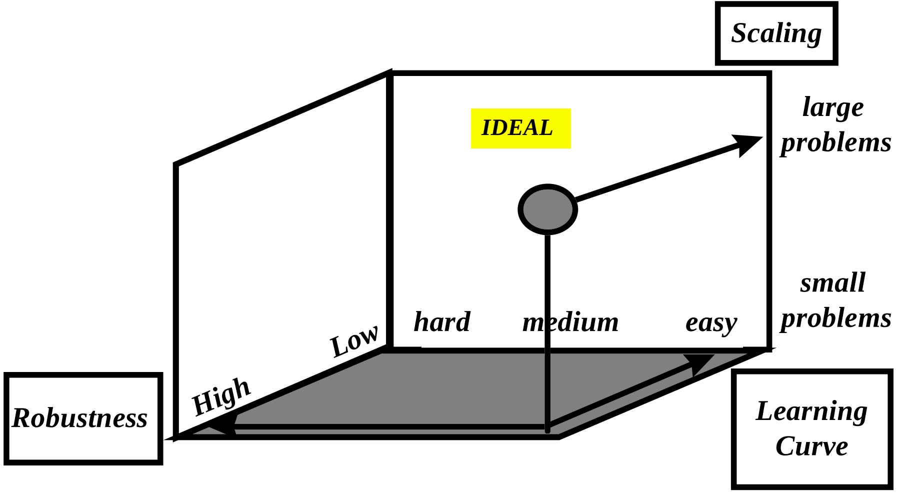
Finally, current CIPT risk management technologies are strongly focused in the cost and schedule arena, but we expect extension to other kinds of risks, such as environmental hazards. In particular, we expect to see improved approaches for managing improbable risks such major pollution events.

7.2. Representative Tools

Figure 7.6 shows three of the factors that are most important in selecting an appropriate optimisation or risk assessment tool.

[25] Stanners, D. and P. Bourdeau (eds). 1995. Europe's Environment: the Dobris Assessment. European Environment Agency, Copenhagen.

Figure 7.6: Optimisation and risk management tool selection factors



An 'ideal' tool would be suitable for both large and small problems – its evaluation time would not become unacceptably long as the problem size increases.[26] It would have an easy learning curve, so that you would not be put off using it. Finally, it would be robust. If a product is robust, it has been proven in a wide variety of circumstances, so that you can have confidence that the solutions you obtain are actually optimal. Of course, cost and availability are considerations. In this section we will look at two optimisation tools and one risk management tool:

- Excel Solver a general-purpose, spreadsheet-based optimiser
- What'sBest! – part of a flexible solver suite
- @RISK for Excel – a risk management and decision support tool.

We complete the section by listing a number of other products available for optimisation and risk management.

7.2.1. Excel Solver™ – General-Purpose, Spreadsheet-based Optimiser

Developed by Frontline Systems for Microsoft, Solver is packaged with Microsoft Excel, and thus users have access to all of Excel's functions. Macros and functions in Excel's Visual Basic can control the optimiser and enquire about the status of an optimisation problem through a series of built-in function calls, making the building of 'turnkey' applications possible. Our ratings of Solver are shown in Figure 7.7 and discussed below.

Learning Curve

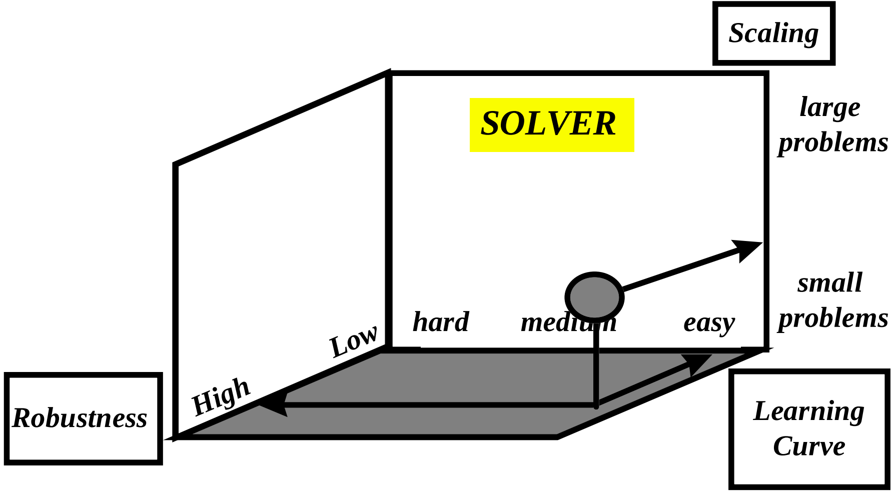
The familiar Excel environment helps makes Solver very easy to use and learn. Example worksheets set up for various optimisation problems are included with Excel, and after playing with these for a short while, one is ready to try it on a small problem. There is on-line help, but surprisingly it provides little assistance for the beginner; it simply lists the commands that are available. A fee-based technical support service helps formulate Solver models for the fastest possible optimisation.

Robustness

Solver uses the 'GRGL' algorithm, which like many others suffers from the property that the solution depends on the starting point. This means that when the algorithm stops, it may have found a locally optimal solution when there is actually a better solution that it has not found.

[26] So you need to project how your ambitions will evolve – an inappropriate product will disappoint you if your challenge grows rapidly, but if your expectations are unlikely to grow greatly, purchasing more power than you need to get started may be a waste. Also, growth in machine power can make up for some lack of scalability.

Figure 7.7: Solver™ selection factor ratings



Scaling
Solver can handle linear programming problems with up to 200 decision variables, and a wide variety of non-linear problems. It is limited by the size of an Excel worksheet, but the package performance is a limiting factor well before the full worksheet size is reached. Most commercial optimisation packages can handle far more variables and constraints than Solver supports. In our view, Solver is useful primarily as a means of learning the technology.

Cost and availability
Solver is available with Excel 4, 5 and 7 running on Windows 3.1, 95, NT and Macintosh. The basic Solver for Excel is included with every copy of Microsoft Excel and Microsoft Office for Windows and Macintosh. Three enhanced optimisers: Premium Solver, Quadratic Solver and Large-Scale LP Solver are available from Frontline Systems *(http://www.frontsys.com)* for problems that exceed the scope of Solver. Table 7.1 shows the other cost factor rankings for Solver, using the categories defined in Table 2.1.

Table 7.1: Cost factor rankings for Solver™

learning	installation	application	price
B about an hour	comes with Excel	B-C hours to days	free with Excel

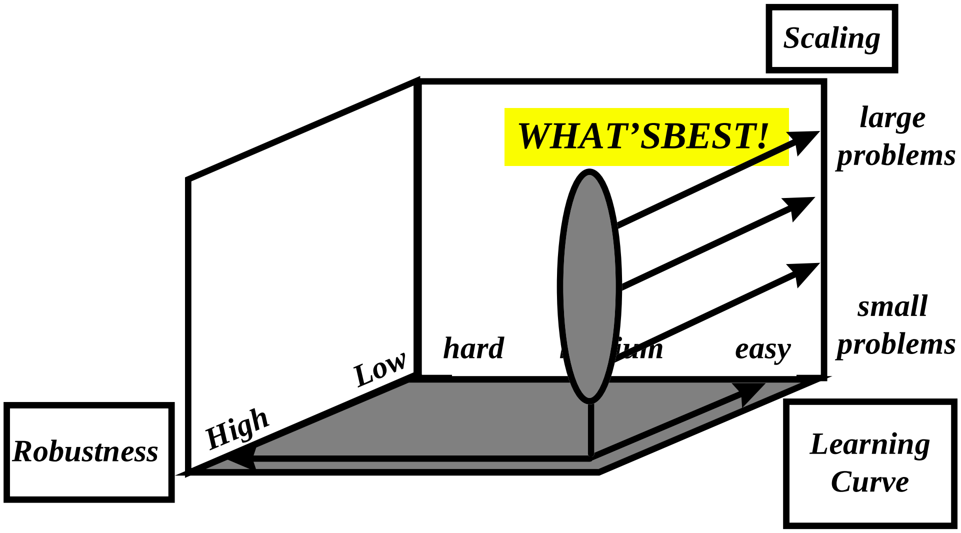
7.2.2. What's Best!™ – Part of a Flexible Solver Suite

What'sBest! is a spreadsheet add-in for use with Microsoft Excel or Lotus 1-2-3. It solves linear, non-linear and integer programs and is a former winner of the PC Magazine Award for Technical Excellence. It can be purchased alone or with two other Lindo products; LINDO, the linear programming solver, and LINGO, the linear, non-linear and integer programming solver, as the Solver Suite. Its ratings are shown in Figure 7.8.

Learning curve
What's*Best!* runs from within Microsoft Excel or Lotus 1-2-3 for Windows and therefore has their menu driven, graphical interfaces. Spreadsheet formulae are used to build the model so no extra programming skills are required. The user does not need to specify which solver to use as this is done automatically. Non-statisticians can therefore be sure of using the appropriate solver for the problem. An Excel Visual Basic interface and a convenient toolbar for Excel users also aid

utilisation. Help is available on-line for all commands and in-depth documentation is provided covering getting started, a brief tutorial and real-world examples from a variety of applications.

Figure 7.8: What’s*Best!*™ selection factor ratings



Users can generate their own models within the spreadsheet using a free-form format. The models accept general and binary integer restrictions and cell ranges can be omitted from the problem formulation. All the spreadsheet functions are available for exploitation in creating the model and pre-processing data. Other features include sensitivity analysis through dual value and range information, recognition of integer and free variables and solution report capabilities. The graphing and plotting capabilities of the spreadsheet are available to display results.

Robustness

What’sBest! solves large-scale linear, non-linear and integer models. The solution process can be interrupted at any time and the best answer at interruption will be given. The solvers are designed for maximum speed and reliability, yet search tolerances and bounds can be set to speed up the solution for integer models.

Scaling

Versions are available to handle a variety of problem sizes, up to tens of thousands of variables and constraints. The rows and columns of the spreadsheet do not limit the size of problem handled, as there is support for multiple sheet workbooks in Excel. The number of variables and constraints, to suit different users’ modelling needs, handled by each of the five versions of What’sBest! are shown in Table 7.2.

Table 7.2: Prices of What’sBest!™ versions

	Suite	*Commercial*	*Professional*	*Industrial*	*Extended*
Variables	200	1,000	4,000	16,000	32,000
Constraints	100	500	2,000	8,000	16,000

Cost and availability

What’sBest! is available from Lindo Systems *(http://www.lindo.com)* in a wide range of versions. It is supported on Windows systems running Microsoft Excel or Lotus 1-2-3. The amount of RAM required will vary depending on the size of the package. Network licences, runtime licences, volume discounts and educational versions are available
from distributors. Table 7.3 shows the other cost factor rankings for What’sBest!, using the categories defined in Table 2.1.

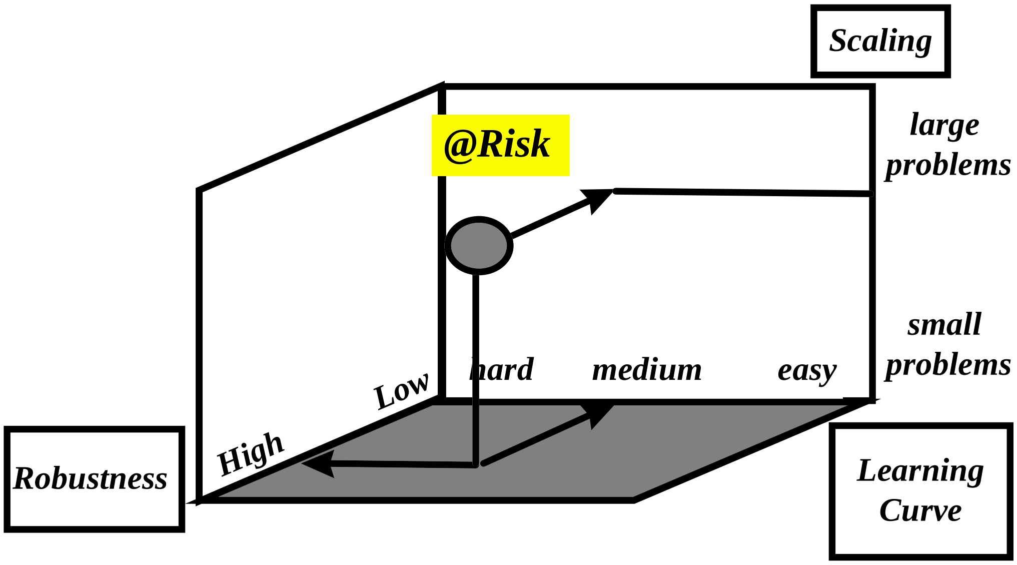
Table 7.3: Cost factor rankings for What'sBest!™

learning	installation	application	price
B about an hour	A less than 10 minutes	B-C hours to days	B-F depending on version

7.2.3. @RISK for Excel™ – Risk Management and Decision Support Tool

@RISK for Excel is by far the most popular tool of its type on the market. It is an Excel add-on, which means that the package is implemented using Excel as a resource. Figure 7.9 shows its selection factor ratings, but before we explore these, we will give you some more information on the product. Referring back to Figure 7.2, the first three panels illustrate the steps of the risk management process with which @RISK for Excel can help you.

Figure 7.9: @RISK for Excel™ selection ratings[27]



Identifying factors
Using Excel, a model of the problem is developed. This stage of the analysis is the same as one does without @RISK for Excel. Let us use an example: deciding whether measures should be taken to improve the condition of sea walls to protect a community against hurricanes. The model will depend on a set of input factors, e.g. value of property in the storm track, time of day, peak storm intensity, sea wall condition, and nominal tide height. Some of these factors may be constant, and others may take on random values from some distribution. Peak storm intensity, time of day, and nominal tide height are examples of random input factors, while property value and sea wall condition are probably constants. You then identify the things you are interested in knowing about: the output factors. As you would for a normal spreadsheet calculation, you enter formulae connecting the input factors (random and constant) to the output factors.

Estimating the range of the factors
This stage may involve consultation with specialists, as Figure 7.2 suggests. For each input factor, a suitable approximation to the probability distribution is chosen. The nominal tidal height distribution is a well measured factor, which could be represented as a normal distribution. The peak storm intensity distribution may be estimated from historical statistics, but with considerable

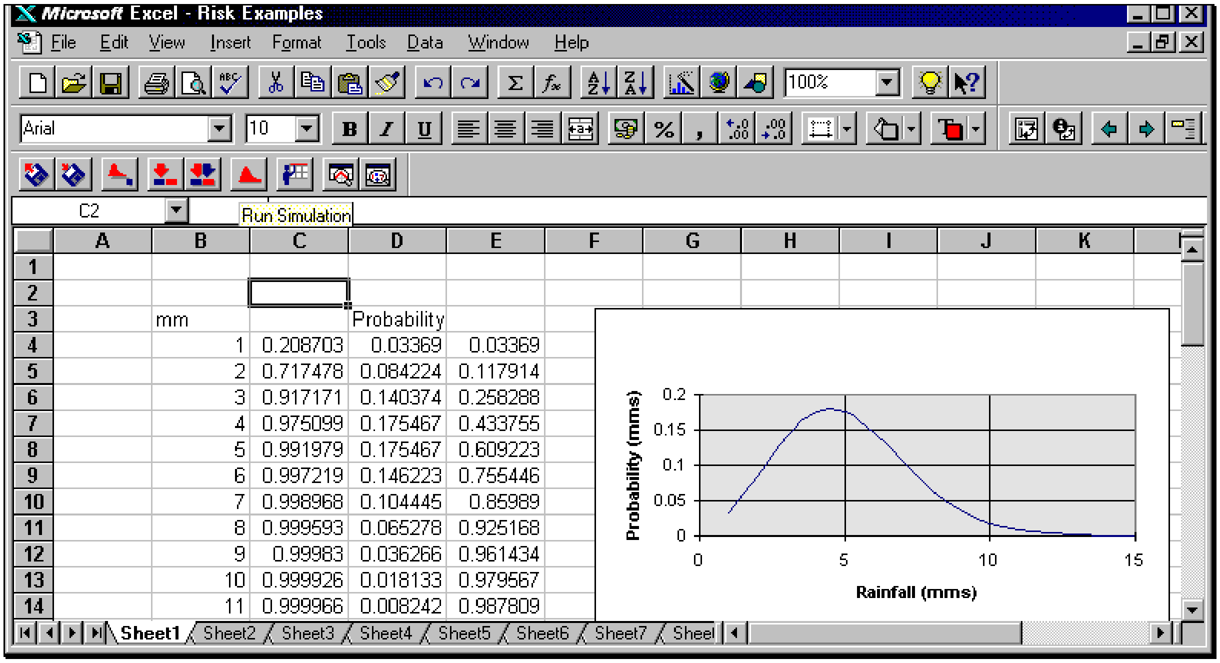
[27] Note: Figure 7.9 should not be compared with Figure 7.7 or Figure 7.8, because @Risk for Excel addresses a different kind of problem

care: just because an event, such as an unusually high wave height or tidal surge, has not occurred this century does not mean that it will not. @RISK for Excel offers more than 30 functions to calculate distributions. In this stage of the analysis, what you do is replace the value of each random factor in your model with the appropriate @RISK for Excel functions.

Analysing the model using simulation

Having set the model up @RISK for Excel now goes to work for you. What it does is quite impressive. What you seek is the distribution function for the output factors, given the variability in the input factors that you have defined. @RISK for Excel calculates this by Monte-Carlo simulation (see Figure 7.10): it runs a large number of trials and accumulates the results of all of the trials into output distributions.

Figure 7.10: @RISK for Excel™ worksheet


Microsoft Excel - Risk Examples
File Edit View Insert Format Tools Data Window Help
100%
Arial
10
C2
Run Simulation
Sheet1 Sheet2 Sheet3 Sheet4 Sheet5 Sheet6 Sheet7 Sheel
0.2
0.15
0.1
0.05
0
Probability (mms)
0
5
10
15
Rainfall (mms)

	A	B	C	D	E
1					
2					
3		mm		Probability	
4		1	0.208703	0.03369	0.03369
5		2	0.717478	0.084224	0.117914
6		3	0.917171	0.140374	0.258288
7		4	0.975099	0.175467	0.433755
8		5	0.991979	0.175467	0.609223
9		6	0.997219	0.146223	0.755446
10		7	0.998968	0.104445	0.85989
11		8	0.999593	0.065278	0.925168
12		9	0.99983	0.036266	0.961434
13		10	0.999926	0.018133	0.979567
14		11	0.999966	0.008242	0.987809

In detail, for each trial, for each of the random input factors, @RISK for Excel chooses a random value from the relevant probability distribution function. Using the spreadsheet formulae you have defined, @RISK for Excel then computes the output factors(s). Each time it computes a trial, @RISK for Excel adds the result to an output factor distribution function. When @RISK for Excel is finished (which may take a few minutes or longer) what you get is a distribution function for each output factor, one that takes into account all the variability in all of the factors you have identified. With this information, you are in a position to judge what is an acceptable risk. For example, you may decide to build a sea wall which will (on average) be overwhelmed only twice in each century. In this case, you will choose a sea wall height that corresponds to a 2% value on the storm height cumulative probability distribution curve. @RISK for Excel then provides tools to help you to identify which factors had the most effect on the results.

Now let's turn to those ratings.

Learning Curve

Experience with the product shows that although users want and need the capability it offers, it is quite difficult for the average manager to understand and use it. The manual is only of average quality, and although there are tutorials, the on-line help is not optimal. It will take some time for most users to take advantage of the advanced features in @RISK for Excel. The manual warns:

> *"Like any tool, (quantitative analysis techniques) can be used to good advantage by skilled practitioners, or they can be used to create havoc in the hands of the unskilled. In the context of Risk Analysis, quantitative tools should never be used as a replacement for personal judgement."*

The message is to practise with @RISK for Excel on toy problems and non-critical projects before undertaking major enterprises.

Robustness

The simulation approach applied in @RISK for Excel is very general, thus you would expect @RISK for Excel to be applicable to a wide range of problems, which it is. It will take advantage of all the available memory, and PCs configured with 64 Mb to do @RISK for Excel calculations have been noted. One class of problems that it is not easy to apply to is planning to meet schedules, and for that a special version, @RISK for MS-Project is offered.
However, for this class of tasks, we would recommend considering other products (for example Risk-Plus), because the @RISK paradigm is not particularly intuitive when working with networks of tasks.

Scaling

@RISK for Excel is limited by the size of an Excel worksheet and the available RAM memory. For many environmental applications, this will not be a problem.

Cost and availability

@RISK for Excel is available from Palisade Corporation *(http://www.palisade.com)* and runs on all platforms that support Excel. Table 7.4 shows the other cost factor rankings for @RISK for Excel, using the categories defined in Table 2.1.

Table 7.4: Cost factor rankings for @RISK for Excel^{YM}

learning	installation	application	price
C one to several days	A-B less than an hour	C one to several days	C

7.2.4. Other Products

Table 7.5 provides summary details of some of the other optimisation and risk management packages that are on the market today.

7.3. Environmental Applications

Typical applications for optimisation tools include personnel or equipment scheduling, raw material or product blending, vehicle routing, financial portfolio management – in summary the allocation of limited resources to meet objectives, while minimising costs or maximising profits. However, the technology is being used increasingly in environmental fields. For example Analytica has been applied in the following areas:

- integrated assessment of the effects, benefits, and costs of acid rain mitigation strategies
- analysis of the cost-effectiveness of pollution-control technologies for fossil-fuel power plants
- policy analysis of the effects of global climate change due to the greenhouse effect.

A number of reference books on optimisation are listed in Section 6.4 and several provide detailed examples using a particular package such as Albright and Winston (1997), Camm and Evans (1996) and Ragsdale (1997), all use Excel Solver. In addition a very useful tutorial is included on the Frontline Systems web-site *(http://www.frontsys.com/tutorial.htm).*

Risk management is at the heart of environmental protection – for example, in assessing the likelihood of pollution from accidents, the likelihood of problems arising from the normal and abnormal operation of industrial processes, in predicting the likely impacts associated with new synthetic chemicals, and so on. A book edited by Calow (1997) has made a very timely appearance with public perception of risks from threats such as nuclear wastes, BSE and E. coli at an all-time high. This new handbook covers the way that scientific methodologies are used to assess risks from human activities, and the resulting objects and wastes, for the environment and for people in the environment.

The understanding of such risks is vital in the framing of legislation, in managing problems in the various major habitats, the practice of management in business, and in managing development programmes. Of an applications nature, chapters in the book cover: assessing risks from chemicals in the environment and from biological introductions for ecological systems; ecological monitoring and epidemiology; human exposure to environmental risks; and risk management in the nuclear industry, for waste disposal, in land uses and treatments, in the exploitation of the seas and for inland waters.

Before going on to describe a selection of case studies using these CIPTs, Box 6.1 provides a hypothetical example to illustrate how optimisation technology could be used, in the future, to levy environmental taxes.

Box 6.1: Optimisation scenario

Manufacturers and operators of equipment are ‘encouraged’, both by legislation and the preferences of customers, to minimise the impact on the environment. The use of advanced optimisation technology is essential for example, to the automotive industry. Recently manufactured automobiles include microprocessors that continuously optimise the fuel/air mixture for driving conditions. In the less market-driven areas of the economy, such as discharge management, optimisation is less common. In this hypothetical usage scenario, we explore how this might change in the near future.

A lake near an industrial area is polluted by discharges from a number of factories. The average annual flow of each type of effluent from each plant is known. The toxicity of the different effluents varies. A linear ‘environmental harm function’ exists for each. The problem is to construct a levy system that reduces the total environmental harm by a defined amount while minimising the total levy charges. We assume that there are just two factories, each discharging annual quantities D1 and D2 tonnes of effluents. The technology works with any number; it’s just simpler to explain with two.

The harm functions for the effluents are defined as H_1 and H_2 damage units per tonne. Thus the total environmental harm (H_{net}) is: $H_{net} = H_1 * D1 + H_2 * D2$, which must be reduced by an amount ΔH. To achieve the reduction ΔH, we propose to impose levies L_1 on D1 and L_2 on D2. What should the values of L_1 and L_2 be?

To keep the example simple, we assume that all companies will reduce the levels of effluents in their discharges in proportion to the levy. The levy is a fee per tonne of discharge. The effect of the levy on the discharge is expressed in the constant α, which has units of the fractional reduction per ecu. Thus if α is 1%/1000 ecu, and we impose a levy of 1000 ecu per tonne, we can expect a 1% reduction in toxin levels in the discharge.

$$\Delta H = \alpha * (L_1 * H_1 * D1) + (L_2 * H_2 * D2)$$

The cost of the levy, which we want to minimise, is: Cost = L_1 D1 + L_2 D2. There are some constraints. Discharges can be zero, but they cannot be negative and (for purposes of this example) levies cannot be negative. Suppose we have the target of reducing H_{net} by 25% annually. The constraint equations are:

$$L_1 (\alpha * H_1 * D1) + L_2 (\alpha * H_2 * D2) = .25*(H_1 * D1 + H_2 * D2)$$

$$D2 - L2 * \alpha * D2 \geq 0; \qquad L1 \geq 0$$

$$D1 - L1 * \alpha * D1 \geq 0; \qquad L2 \geq 0$$

Given this information, and values for the constants, a linear programming package will identify the optimal values of L_1 and L_2.

This of course is a very simplified example; the environment is complex – synergistic effects and thresholds abound. However, it is very worthwhile to work hard with LP to see if the problem at hand can be represented in this way. Often variable transformations and other tricks make it possible to do so. This then allows you to take advantage of well understood, guaranteed technology. If it is not possible to express the problem linearly, then non-linear optimisation technologies can be brought to play. Be warned however; these are not guaranteed to find optimal solutions in every case.

Now we refer to five case studies, all projects currently under development, which are looking to introduce optimisation and risk management strategies to operational procedures. The first case study is in the energy production sector, the next three concern water management and the final case study involves waste treatment.

Energy production

The European Commission is involved *(http://rocks.worldbank.org/html/fpd/em/power/EA/methods/mtesmeme.stm)* in an assessment of methods and tools for environmental management in the power sector. Dynamic optimisation models (EFOM-ENV models) are being developed, employing linear programming, to represent the energy producing and consuming sectors in each Member State. The models optimise the development of these sectors under given fuel import prices and useful energy demand over a pre-defined time horizon. The development of national energy systems can be subject to energy and environment constraints like availability of fuel supply, penetration rates of certain technologies, emission standards and emission ceilings. The model data bases contain a wide range of end use technologies such as conventional technologies, renewables, efficient fossil fuel burning technologies, combined heat and power and energy conservation measures in the demand sectors. Inputs to the models include: energy balance in a base year; projections of useful energy demand and fuel prices, investment and operating costs; availability and efficiency of energy production, transformation and the technologies; energy and environmental constraints, where relevant (e.g. development of nuclear power, availability of renewable resources, emission standards, etc.); emission factors (SO_2, NO_x, particulates, CO_2); and load duration curves for electricity and heat demands. The basic version of EFOM-ENV, used by twelve Member States, is in FORTRAN and runs on main- frame computers. However, there is an OMNI version of the Belgian EFOM-ENV model and a GAMS version, for the Netherlands, Poland and the Czech Republic – these versions run on PCs.

Water management

Water distribution networks are geographically distributed systems, with considerable heterogeneity in terms of control structures, management strategies, and varying geometry with continuous expansion and changes in demand. Due to these characteristics, water distribution companies face the problem of data and knowledge integration related with control and optimal exploitation. A number of European projects are currently focused on the design and development of a next generation system to support the control, optimal operation and decision support of drinking water distribution networks. One such project is WATERNET, an evolutionary knowledge capture for the advanced supervision of water distribution network *(http://webserver.tagish.co.uk/ethos/tap/partners/2212.htm)*. The WATERNET System (see also Camarinha-Matos and Martinelli, 1997 and Afsarmanesh et al, 1997) assists the distributed control of water management networks to minimise the costs of exploitation, guarantee the continuous supply of water with a better quality monitoring, save energy consumption and minimise natural resources waste. This system comprises several subsystems: a distributed information management subsystem, a machine learning subsystem, an optimisation subsystem, a water quality monitoring subsystem, a simulation subsystem, and a supervision system that integrates these subsystems in order to assist the decision making and optimal operation of the network. The basic raw material for this work are historical data from a Portuguese water distribution company that has 45 water stations and some of then with 6 years of collected data taken every 5 minutes.

Another water industry project using optimisation tools is WATERCIME, which aims to develop a standardised, cost-effective and environmentally sensitive tool, which will increase the implementation and efficiency of existing and future water management projects and which is based on a multi-vendor environment. The work is focusing on security, public health, environmental and operating costs issues *(http://www.laas.research.ec.org/esp-syn/* text/8399.html).

TAP-EXTRA (Trial APplication of EXplaining Technology for Risk Assessment and monitoring) is a European Commission Fourth Framework project to take the technology of enhancing complex software with co-operative, explanatory behaviour, and apply it to an application for the control of the rainstorm network of Bordeaux *(http://apollo.cordis.lu/cordis-cgi/srchidadb?ACTION=D&SESSION=189641997-12-1&DOC=1).* This will allow a new class of users, less expert in the domain, to make use of the application and will support confidence building and training with more experienced users. The project is directed at automated control and optimisation techniques for complex and possibly safety critical systems. Industrial applicability is widespread: there are applications with similar characteristics in the water industry, and other industrial applications where there are requirements for operators to monitor complex data in real time, where expertise is at a premium and where there are business benefits from opening the application to new classes of users.

Waste treatment

On the waste treatment side of the industry Acer Environmental has developed a sewage sludge disposal model in response to the need for a strategic planning tool to evaluate the most cost-effective option for sludge treatment, transport and disposal *(http://www.dash.co.uk/applics1.html#sewage).* XPRESS-MP software was used to build a model using input data and then to optimise it. The input data includes: source identification, national grid references, sludge quantity, sludge quality and potentially toxic elements (PTE), disposal destination identification, capacity, constraints, costs, treatment performances, available transport routes and distances. The output from the model, DISPEL, details the total operating cost and cost breakdown (transport, treatment and disposal), the optimised routes for sludge transport from source to sludge treatment centre and destination, and the spare capacities (solids, volume and PTE) of both sludge treatment centres and destinations. Output can also be produced via a GIS, which provides result presentation in the form of maps showing sludge transfers and positions of sources, works and destinations. DISPEL has been successfully used strategically to show which sludge treatment centres and disposal sites are under-utilised and which are fully utilised, as part of a number of sludge strategies undertaken for various water companies.

Table 7.5: Other optimisation and risk management products

Product	Overview	Features	Data limits	Platform	Ease of use	Comments
Analytica™	A visual environment for creating, analysing and communicating probabilistic models for risk and decision analysis. The successor to Lumina's Demos decision modelling system.	Hierarchical influence diagrams. Uncertainty about any variable is expressed by selecting a probability distribution. Uncertainties are propagated through the model using Latin hypercube or Monte-Carlo sampling or uncertain results are displayed as standard statistics, probability bands, probability density functions, or cumulative probability functions.	14,900 variables, 15 dimensions per array, 32,000 elements per dimension, 32,000 sample size.	Macintosh 68020 and later; 2.5 Mb RAM for 68000 version; 4 Mb RAM for Power Macintosh version, System 6.0 and up. Developers can build and distribute applications which access the Analytica Decision Engine on Windows 95 and Windows NT platforms. Analytica ported to the PC platform in 1997.	Uses a small number of standard diagram symbols making it easy to use and learn. Buttons and nodes are simply clicked to evaluate a model and nodes positioned and arrows drawn between them to create a graphical model as an influence diagram.	From Lumina Decision Systems (*http://www.lumina.com*). Demonstrations are available for viewing or downloading.
BestFit	Part of the Palisade DecisionTools Suite. For distribution fitting.	Identifies which of the 28 probability distribution types and which parameters fit the data best. Data read directly from spreadsheet and results output as graphs and reports.	Up to 30,000 data points or pairs.	Windows 3.1 or higher. Minimum 4 Mb RAM. Works with, but does not require Excel or Lotus 1-2-3.	BestFit's wizard takes users through the set up of a run step-by-step.	From Palisade Corporation (*http://www.palisade.com*).
CPLEX Linear Optimizer Base System	The linear optimiser is the foundation for all CPLEX software. Fast and robust primal simplex, dual simplex and network simplex solvers for linear programming problems.	Problem pre-processing, reporting, messaging control, interactive revision capability, sensitivity analysis and a simple command structure. Enter problems directly or from files, view statistics or histograms. Automatically determines smart setting of algorithm parameters or can be adjusted manually.	Handles problems of unlimited size and difficulty. Has been tried and proved with millions of variables and constraints.	PCs: Windows 3.1, 95 and NT, DOS, OS/2, Power Macintosh; UNIX: mainframes. Requires 2 Mb of hard disk space, 2 Mb RAM (varies with the size of problem).	Easy-to-use, user interface with on-line help. Technical support by phone, fax or e-mail; licence transfers; prior purchase credits for new licences; newsletter subscription.	From CPLEX Optimization Inc (*http://www.cplex.com*). Other CPLEX products include a Callable Library, Mixed Integer Solver, Barrier/QP Solver and Parallel Solvers.
Evolver	The genetic algorithm-based optimiser for Microsoft Excel.	Choice of six genetic algorithms for solving linear, non-linear and complex problems.	Limited by the number of rows and columns of the spreadsheet.	Windows 3.1, Windows 95 and NT, running Excel 4, 5, or 7.	Runs from Excel spreadsheet, so interface should be familiar to most users.	From Palisade Corporation (*http://www.palisade.com*). Includes tutorial and templates.
GAMS	Genetic Algebraic Modelling System, a high level modelling language for formulating models. Is independent of solution algorithms of specific solvers.	Handles linear, non-linear, mixed integer linear, mixed integer non-linear and mixed complementary problems.	The student version is limited to 1000 non-zero elements in the constraint matrix, of which 200 can be 'non-linear non-zeros'. Discrete variables are limited to 20.	Windows 3.1, NT, 95, DOS 6.0 or OS/2; UNIX: SunOS, Sun Solaris, DEC Ultrix, DEC Alpha UNIX, SGI IRIX, HP-UX, OpenVMS; VAX/OpenVMS, DEC Alpha Open VMS.		From: GAMS Development Corp (*http://www.gams.com*).

Product	Overview	Features	Data limits	Platform	Ease of use	Comments
GeneHunter	Combines a Microsoft add-in with a programmer's tool kit of genetic algorithm functions to provide a development environment. Callable from Excel 5 spreadsheets or via function calls in a Dynamic Link Library (DDL).	A problem is entered in a dialogue box. The Fitness Function box gives the location of the cell containing the formula which measures the success in finding a solution to the problem. This can be a mathematical formula, neural network or logical rules. Only one Fitness Cell is allowed but sub-goals can be used to find an optimal solution. Chromosomes are variables whose values are adjusted to solve the problem.	Runs ≤128 populations simultaneously, each with ≤2,000 individuals. Uses ≤5,000 integer/non-integer chromosomes in one individual or ≤2,000 enumerated chromosomes. Create ≤ 200 chromosome pools in one population, each pool with ≤10,000 chromosomes.	Windows 3.1 or above and Excel 5 or later. Versions for both 16-bit and 32-bit operating system are included. 80386 or higher coprocessor with 4 Mb RAM and 1.5 Mb hard disk space.	GeneHunter can be run from the Excel interface. Once the data has been entered and the problem defined simply press *Start* for the calculations to begin. Advanced features are available for programmers.	From Ward Systems Inc (*http://www.wardsystems.com*). A free runtime license with limited restrictions is available for programmers. GeneHunter can be integrated with NeuroShell and NeuroWindows neural network packages.
OSL	Library of optimisation routines for linear, mixed integer and quadratic programming.	The PC version includes the subroutine library and a set of stand-alone OSL application programs. The host and workstation OSL can be called from FORTRAN, PL/I, C and APL2. Calls high- or low-level subroutines. A MOTIF based GUI is available for IBM RISC System/6000 workstations providing a point and click interface.	There are no inherent limits on the size of the problem that can be handled, provided enough memory is available.	IBM-compatible PCs with 386, 486 and Pentium micro-processors running DOS, OS/2, Windows 3.1 or Windows NT. A variety of mainframe and workstation environments. Minimum of 8 Mb memory.	OSL is usually called from a programming language and is therefore unsuitable for non-programmers. The PC version also uses command-line options. The learning curve for this package is steep.	From IBM (*http://www .research.ibm.com/osl/*). A student version is available which includes 90% of the functionality of the full library.
Quattro Pro Solver	A general-purpose optimiser similar to the Excel Solver.	For linear, integer and non-linear programming problems. Solver models can not be exchanged with Excel or 1-2-3.	For small-scale problems.	Windows, Windows 95 and NT running the Quattro Pro spreadsheet.	Programmable only through keystroke macros so it is a little less easy to use than Solver for Excel or 1-2-3. The user interface is slightly different to the Excel Solver.	Corel Corporation (*http://www.corel.com*). Is included free with the Quattro Pro spreadsheet in Novell PerfectOffice (known as Corel WordPerfect Suite)
Risk+ for MS-Project	Adds to MS-Project ability to specify range and distribution for schedule drivers. Simulates and displays risk curves.	Schedule and cost risk analysis, tightly integrated with MS-Project.	Limited by MS-Project.	Windows 95. Requires MS-Project.	Cue cards and on-line help.	From Palisade Corporation (*http://www.palisade.com*).
@RISK™ for MS-Project	Adds @RISK capabilities to MS-Project.	Monte-Carlo or Latin Hypercube simulation of network. View probability of any task on the critical path.	Limited by Excel and MS-Project.	Windows 95. Requires Excel and Project.	Somewhat non-intuitive links to @RISK environment from MS-Project.	From Palisade Corporation (*http://www.palisade.com*).
RISKview™	Tool for viewing, assessing and creating probability distributions.	Comes in two versions, RISKview and RISKview Pro. Allows quick assessment of probabilities when building models. Distribution curves can be drawn free-hand.			RISKview 'pops-up' over @RISK for Excel, Lotus 1-2-3 for Windows, or MS-Project to instantly display any distribution in an @RISK model.	From Palisade Corporation (*http://www.palisade.com*).

Product	Overview	Features	Data limits	Platform	Ease of use	Comments
TopRank	For What-if analysis. Part of the Palisade Decision Suite.	Automatically identifies and ranks the important variables which affect spreadsheet results. Results are summarised in tables, tornado, spider and sensitivity graphs.			Appears as a toolbar in your Excel or Lotus spreadsheet.	From Palisade Corporation (*http://www.palisade.com*).
XPRESS-MP for Windows	Modeller and optimiser are available as subroutine libraries which can be linked into end-user applications.	Mixed Integer Programming (MIP) optimiser and Newton Barrier Integer Point optimiser for very large problems are available. Includes an intelligent text editor, problem database and keyword insertion capability. Available as Dynamic Link Libraries (DDL). Uses ODBC to connect to spreadsheets.		PCs under DOS, Windows and Windows 95 and Macintosh. Workstation versions available for Sun SPARC, HP, Silicon Graphics, DEC, RS6000, NEC, VAX, DEC Alpha and for mainframes: VM/CMS and MVS.	The familiar Windows environment makes this easy-to-use for novices. The optimiser is accessed through menu selections and dialogue boxes. Context-sensitive on-line help and training is available.	From Dash Associates (*http://www.dash.co.uk*). A demo and evaluation copy are available from the product web site. Free updating of XPRESS products is available. An educational version of XPRESS-MP for Windows retains all the functionality of the commercial product.

7.4. References and Bibliography

Afsarmanesh, H., L. M. Camarinha-Matos and F. Martinelli. 1997. Federated knowledge integration and machine learning in water distribution. In: Re-engineering for Sustainable Industrial Production, edited by L. M. Camarinha-Matos. Chapman & Hall.

Albright, S. C. and W. L. Winston. 1997. Practical Management Science: Spreadsheet Modeling and Applications. Duxbury Press.

Calow, P. (ed). 1997. Handbook of Environmental Risk Assessment and Management. Blackwell Sciences Ltd.

Camarinha-Matos, L. M. and F. Martinelli. 1997. Application of machine learning in water distribution networks: an initial study. In: Proceedings of the Workshop on Machine Learning : Application in the Real World; Methodological Aspects and Implications, edited by R. Engels et al. Nashville, USA.

Camm, J. D. and J. R. Evans. 1996. Management Science: Modeling, Analysis and Interpretation. South-Western College Publishing.

Chapman, C. and S. Ward. 1997. Project Risk Management. Wiley & Sons.

European Environment Agency. 1996. Environmental Taxes Implementation and Environmental Effectiveness. Environmental Issues Series, No. 1. Office for Official Publications of the European Commission, Luxembourg. (see also: *http://www.eea.dk/frdocu.htm*).

Fletcher, R. 1991. Practical Methods of Optimization. Wiley & Sons.

Grey, S. 1994. Practical Risk Assessment for Project Managers. Wiley & Sons.

Hung, M. S., W. O. Rom and A. D. Warren.1993. Optimization with IBM OSL. Boyd & Fraser).

Jakeman, A .J., M. B. Beck and M. J. McAleer (eds). 1993. Modelling Change in Environmental Systems. Wiley & Sons.

Luenberger, D. G. 1984. Linear and Nonlinear Programming. Addison Wesley.

Psarras, J., P. Capros and J-E. Samouilidis. 1990. Multicriteria analysis using a large-scale energy supply LP model. European Journal of Operational Research, 44, 383-394.

Quevedo, J., G. Cembrano, A. Valls and J. Serra. 1988. Time series modelling of water demand. A study on short-term and long-term predictions. Computer Applications in Water Supply. Wiley & Sons, 146-164.

Ragsdale, C. T. 1997. Spreadsheet Modeling and Decision Analysis: A Practical Introduction to Management Science. South-Western College Publishing.

Remmers, J. et al. 1990. Integration of air pollution control technologies in linear energy-environmental models. European Journal of Operational Research, 47, 306-316.

Schrage, L. 1997. Optimization Modeling with LINDO. Duxbury Press.

Taylor, D. M., S. E. Metcalfe and J. Hardisty. 1996. Computerised Environmental Modelling – A Practical Introduction Using Excel. Wiley & Sons.

Ulanicki, B. and C. H. Orr. 1990. An optimization technique for water supply and distribution systems, control theory and advanced technology (C-TAT). Mita Press.

Vose, D. 1996. Quantitative Risk Analysis. Wiley & Sons.

8. INTELLIGENT AGENTS

This chapter is slightly different from the previous ones, in that it describes technology that is just about, but not quite, ready for use. It is included here so that you will be ready to take advantage of what it offers as it emerges.

8.1. Capabilities and Limitations

What are intelligent agents?
One of the most interesting of the new trends in computing is 'Intelligent Agents'. The core idea is to have software units called agents that can perform high-level Internet-related tasks, such as researching a topic. An agent is given a task to perform, and it does the work keeping in mind the desires and preferences of the person. Reflecting the current state of the art, agents are sometimes represented as puppies – eager to please but not particularly intelligent. How is an agent different from an ordinary program, such as a web browser, word processor, or neural network?[28] The main difference is this: an ordinary program is passive unless it is responding to precise instructions you give it (open file, insert a character). An agent is shown what you want to achieve, rather than how to achieve it, and then it uses its own initiative to meet this objective. This means that it can accomplish much higher-level tasks than ordinary programs. There are two other important features of agents:

- They are personalised: they know not just what you want to achieve, but also something about your preferences and dislikes. So the solutions they deliver are compatible with your values.
- They adapt, both to new information you provide and to new information they encounter as they seek to achieve your goals.

Why is agent technology important?
It's important because, in the midst of an information explosion, the effort it takes to gather information is becoming a significant factor. The next paragraphs expand on this point.

There is an accelerating trend to put information of all kinds on the 'web'. The search engines that go to work when you push the button on your browser make it is easy to find relevant documents, but they also return a very high proportion of information that is not of interest. The efficient handling of information from multiple sources has been a problem for sometime, but the web has made the problem more difficult, for three reasons:

- a vast number of different information resources
- constant updating of information
- the lack of central control.

Agent technology is believed by many to be an important element of the solution to this problem. However, it must be admitted that not everyone agrees, and there are competing ideas. The web is perhaps the first arena that agents can help environmental analysts, but it is far from the end of the story. The agent concept is believed by many to be the 'grand idea' that will link all of the world's computer systems into a giant intelligence amplifier for humanity.

Agent buzzwords
Here is some agent jargon:

[28] The features described in this section were listed by Patti Maes in her presentation 'Software Agents' at the Second International Conference on the Practical Application of Intelligent Agent and Multi-Agent Technology, London, April 21-23, 1997.

Broker: *an agent that helps other agents, by providing connections, translations, or other similar services.*
Data mining: *searching in databases for undiscovered relationships, also known as exploratory data analysis.*
Format: *the way information is represented when it is stored, for example as words or numbers, and how many decimal places.*
Metadata: *information about information – for example, what kind of data is in a dataset.*
Schema: *the way information is organised in a database.*
Ontology: *the set of terms and definitions that apply in some domain; how meaning is expressed.*

8.1.1. Key Capabilities

The key capabilities that intelligent agent technologies offer environmental knowledge workers are:

- efficient ways of finding relevant information on the Internet
- tools to automatically invoke appropriate information services, such as database queries, data processing and knowledge systems
- an ability to merge information from multiple sources to handle complex service requests.

These three capabilities are ordered to indicate what is available now, what we can expect to see in the near future, and what is hoped for, but still seems some way off. Let's take a look at each of them now.

Efficient ways of finding relevant information on the Internet
There are already a number of products that serve this niche. They are just crossing the threshold from interesting demonstrations to useful aids. With only a short period of familiarisation, agents can be configured to perform in-depth topic searches. This is especially efficient when they are set up to run in the user's rest periods, because they can consume significant machine resources. The quality of the results that this class of agents delivers is rather variable; we have had most success when we monitored the progress of the agent now and then and helped it along a little. However, there is a problem. Increasingly, web-based information providers are not storing HTML pages but are constructing pages on demand and customising them according to information about the user. Where this occurs, direct links are missed. This problem applies not just to agent-based search tools, but also to the indexing performed by all the popular search engines.

Tools to automatically invoke appropriate information services
This capability is offered in some prototype systems (e.g. Infosleuth, see Section 8.2.2), but is not yet distributed commercially. Many environmental analysis tasks demand that information from multiple sources be retrieved and integrated. For example, compiling a European-wide summary of the state of the environment, which requires access to a large number of different national and regional datasets. When these tasks must be performed repeatedly, automation could be worthwhile, but only if the costs are justified. Agent technology promises to reduce the cost of retrieving information from multiple sources[29].

The reason that this is not at all a trivial task has to do with three topics: formats, schema and ontologies. If a information system 'dictator could write a set of rules, saying that all environmental data had to be stored in a particular format, using a particular schema, and according to the definitions of a particular ontology, and also preferably in a particular database

[29] The first step in establishing this capability is to set up a mechanism through which datasets and services can be advertised and accessed. Two examples of this are the United States Environmental Protection Agency's Environmental Data Registry *(http://www.epa.gov/edr/)* and the European Commission's Centre for Earth Observation *(http://www.ceo.org/faq.html)*.

product running a particular kind of computer, then merging information from multiple sources would not be nearly as difficult as it is. Unfortunately, in every one of these aspects, there is diversity. The hardest aspect of the problem is the ontologies. What people actually mean by the information they record can differ in numerous and subtle ways. Some leave out information that others consider crucial. It is in this area that progress is most needed, which is why the Infosleuth and EDEN projects (see Section 8.2.2) are so important.

Merging information from multiple sources to handle complex service requests
This capability should become available within the next five years, if not sooner. The next stage after recovery of information from multiple sources is merging this information appropriately. What 'appropriate' means depends, of course, on what the user wants to accomplish. An everyday example is travel planning. Meetings must be arranged; airline, car and hotel reservations must be organised; itineraries, including places to dine and visit, must be prepared. The hotel and conference venue should not be distant from one another, and the choice of whether or not to hire a car depends on three factors: personal preferences, cost and the quality of the local public transport. 'Trivial' tasks such as this can consume a lot of time; they are difficult even for human travel agents to perform. The reason is there are a number of decisions that need to be made that depend on personal preferences, and optimising for these preferences is complex. The agent solution is the Personal Agent – software that knows your preferences and negotiates on your behalf with agents representing hotels, conference organisers and others.

The challenges we give our agents in the environmental domain may turn out to be much more complex than arranging travel; they might involve querying legal databases, extracting environmental records, and applying reasoning tools to identify possible problems and/or remedial measures. Nevertheless, the hope is that agent technology will be up to the job.

8.1.2. Limitations and Likely Evolution – the Authors' View

Earlier, we used the phrase *'intelligence amplifier'*. The way agents will eventually act as intelligence amplifiers (we predict) is by allowing your agent, which knows what you want to accomplish, but not how to achieve it, to enlist the help of other agents that can solve different parts of your problem. What we see today, in agents that search for information on the web, is just the beginning. A feature of agent technology that, in principle, makes it attractive for environmental analysts is scalability; the ability to handle both small and large problems. The idea is that complex tasks are broken into smaller pieces, that are handled by a number of agents running on different machines. However, current research does not make a convincingly case.

What is wrong with today's agent technology?
Cynics define today's software agents as 'adolescent objects on a network, with a credit card and an attitude'. Our view is that this is overly harsh. The best of the current generation of web-searching agent software is easy to use and delivers a useful result. However, it is not always clear that employing the agent is more efficient than driving search engines directly – firstly, people are better at getting one or a small number of results quickly, although agents can (sometimes) be more thorough and secondly, agent programs can be greedy, consuming lots of processor and memory space.

At the other end of the spectrum from these simple web-searching agents are large projects such as Infosleuth (see Section 8.2.2). Infosleuth will attempt to apply agent concepts to integrate knowledge from multiple diverse and evolving sources, capturing knowledge in an AI crisp knowledge representation language (KQML). Our analysis of Infosleuth identifies two significant barriers that stand between the Infosleuth and the information agent services that the environmental community needs. These are (1) the practical aspects of translating and maintaining the vast and dynamic knowledge base, and (2) the risk that crisp AI concepts will be found inappropriate for environmental problems. Much of today's agent development is about co-operative problem-solving in domains such as telecommunications and the web – domains that have hard facts (yes/no states), textual information and crisp rules. In contrast the nature of the information required to support environmental decision-making is likely to be very different We see that this information as:

- highly numerical
- inherently subject to uncertainties (fuzzy or probabilistic)
- strongly spatially and temporally organised
- a mixture of facts, theory-based knowledge, beliefs and preferences.

Such aspects will require a significant evolution beyond today's agent technologies before the large-systems approach can deliver a major benefit to environmental workers.

It is impossible to predict how effective Infosleuth will be in the long run, but it would be optimistic to expect that all of the problems of environmental information access will be resolved by it. Nevertheless, in pointing the way to an agent-based solution, and in particular focusing on the ontology aspects, this project is likely to be a major milestone on the way to the flexible environmental information systems required for the 21st century.

What is happening?
Agent technology is an active research area. Some of the topics being explored include:

- operational support and diagnosis
- electronic commerce
- manufacturing
- information finding and filtering
- planning and resource allocation
- process control
- service integration.

The Internet Resources for the Environmental Industry *(http://www.enviroindustry.com/resources.html)* provides a useful web-site where information about the current state of development of Internet technology for the environment can be found.

There are many companies developing and deploying agent-based software. Almost all of these are inexpensive 'personal assistant' products that help manage information on the web. Another active area of product development is shopping agents. For example agents are available that can identify the least expensive source of CDs (from among those advertised on the web). The key ideas being developed for agents are intentionality and co-operation. While object-orientation is the main wave in software today, the next generation agents are being described as 'objects with intentions'. While objects 'know things and can do things' agents are being given the capability to co-operate with other agents to achieve goals.

A key unsolved issue is that of blame and trust. If my 'health and fitness' agent is too eager and overloads the information systems of a hospital in its attempts to protect me, perhaps other people will die. Who is to blame? Me, for training the agent badly, the manufacturer of the agent for not ensuring that things like that cannot happen, or the hospital, for not protecting itself properly.[30] In our opinion, this problem is deep and will make it difficult to have agents that are totally independent. Instead, people will have to be involved, exercising control and taking responsibility for their agents. That is the blame side of the problem. On the trust side, if an agent can spend your money, you want to be pretty sure that the people it is negotiating with are honest. One way this can be achieved is through 'integrity brokers' – agents that take responsibility for ensuring that the agents your agent contacts through them are honest, or at least insured. Of course, this

[30] This issue was raised by Rasmussen (1997), who proposed a co-operative agent-based solution that assumed that agents could identify and remember the behaviour of one another. This does not seem a realistic possibility.

comes back to humans establishing trust with one another. So while agents may make life more convenient, they will not take away responsibility.

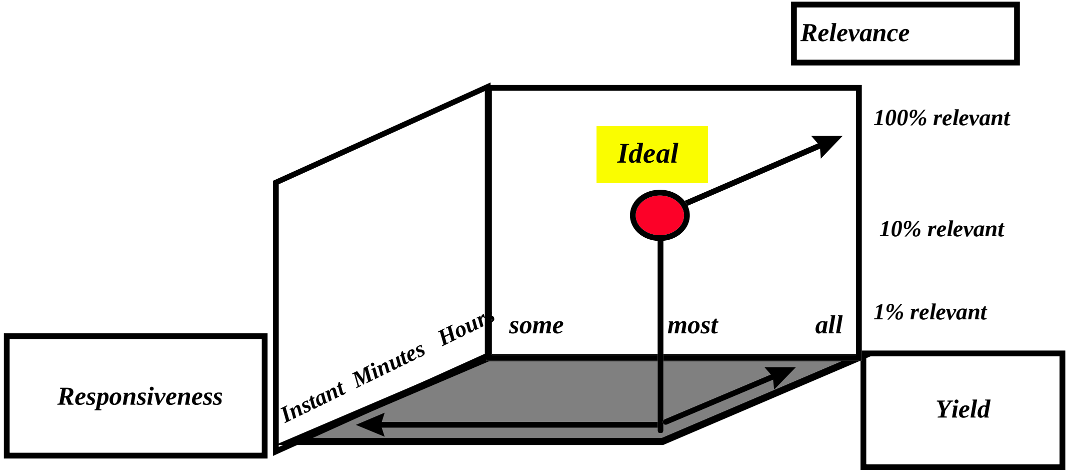
Where is it heading?

The ideal world that agent researchers envision is one in which anything that could be done electronically, can be done that way. Agent-based shopping and information gathering is just the beginning of this grand vision. The most interesting work relates to co-operative real-time problem solving, which could be quite important for environmental analysts in the medium term. The issue is that as the time scale on which decisions need to be made shortens, the luxury of gathering all of the information in one place disappears. An example is a Chernobyl-type disaster. Much data about what happened emerged long after it was needed. In future disasters (probably not the next one!), data access agents will be enlisted by higher-level agents that will have goals of protecting public safety (first goal) and protecting the environment (second goal). They will know about transport networks and the weather conditions, and will make a list of sensible proposals for action to minimise the impact of the problem. More immediately, agents will monitor what is going on in critical environments, and guide people away from wrong decisions.

8.2. Representative Tools

This section surveys the agent-based information retrieval tools. Figure 8.1 shows three of the factors that are most important in selecting an intelligent agent to assist you in gathering information.

Figure 8.1: Web-based agent selection factors



An ‘ideal’ agent would respond instantly, deliver all the relevant information, but no irrelevant information. Some other important factors are:

- learning curve – how easy or difficult it is to learn how to use the tool
- ease of use – how quickly you can do the job once you know how
- cost and availability.

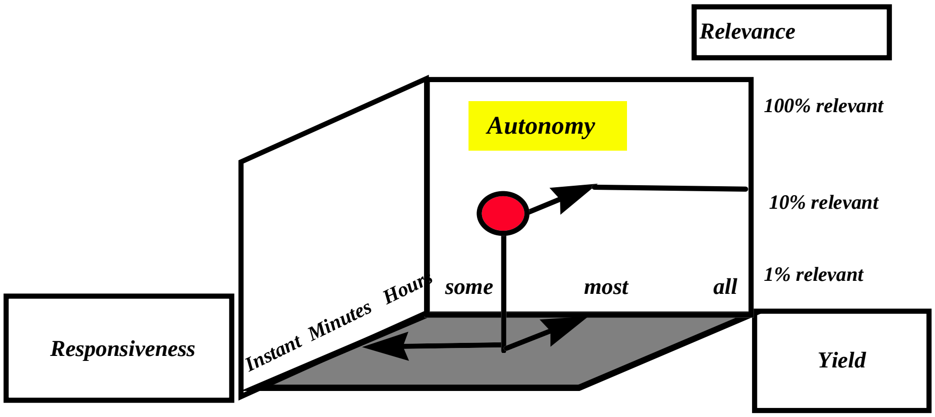
8.2.1. Autonomy Web Researcher™

Autonomy Web Researcher[31] is one of the modes of Autonomy Agentware, the mode that searches for information on the WWW. Other Agentware modes search images by content, prepare customised newspapers, answer questions on a subject, and find other agents with similar interests. We focus on the web searching mode because all of the principles of the technology are illustrated here, and because this is the mode you are likely to find most useful.

[31] Much of the material in this section is adapted from the Autonomy application and on-line help (used with permission).

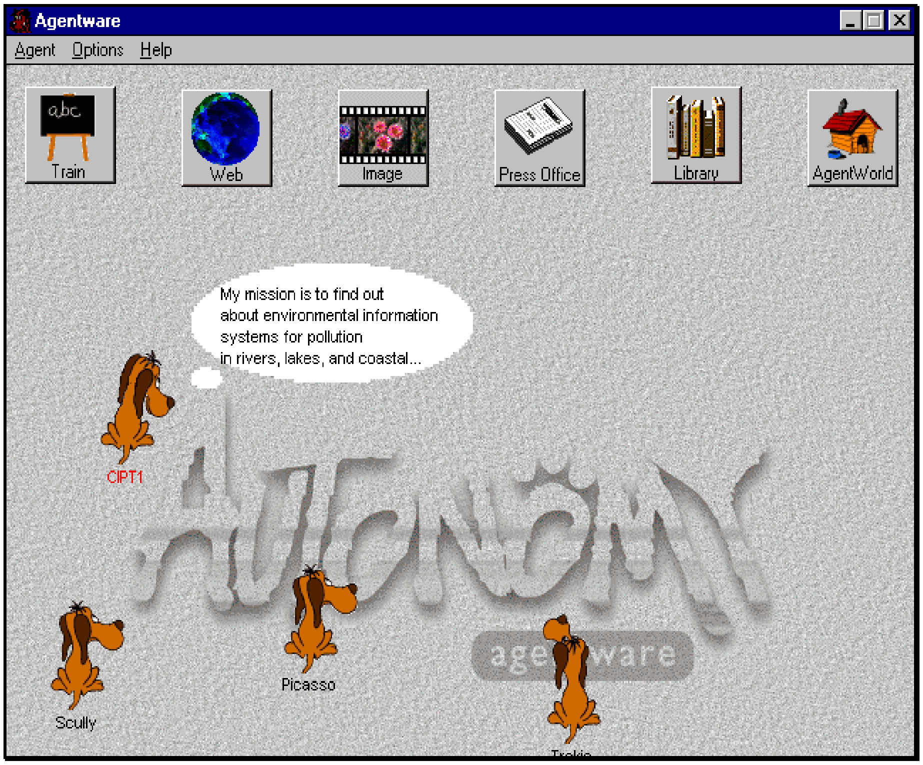
As shown in Figure 8.2, with Autonomy you can expect to retrieve a higher fraction of useful material than with a regular Web search, but there is no guarantee that you will get it all. Results are delivered in minutes, comparable to web searching. On the other factors, we also rate the product highly. It is easy to learn to use the tool; and once it has been mastered, it is efficient with your time (but see below).

Figure 8.2: Autonomy Web Researcher™ selection ratings



To begin working with Autonomy, you create a new agent and give it a mission Figure 8.3. You do this by typing a sentence or two that describes what you want to know. Doing this well is a skill acquired through experience.

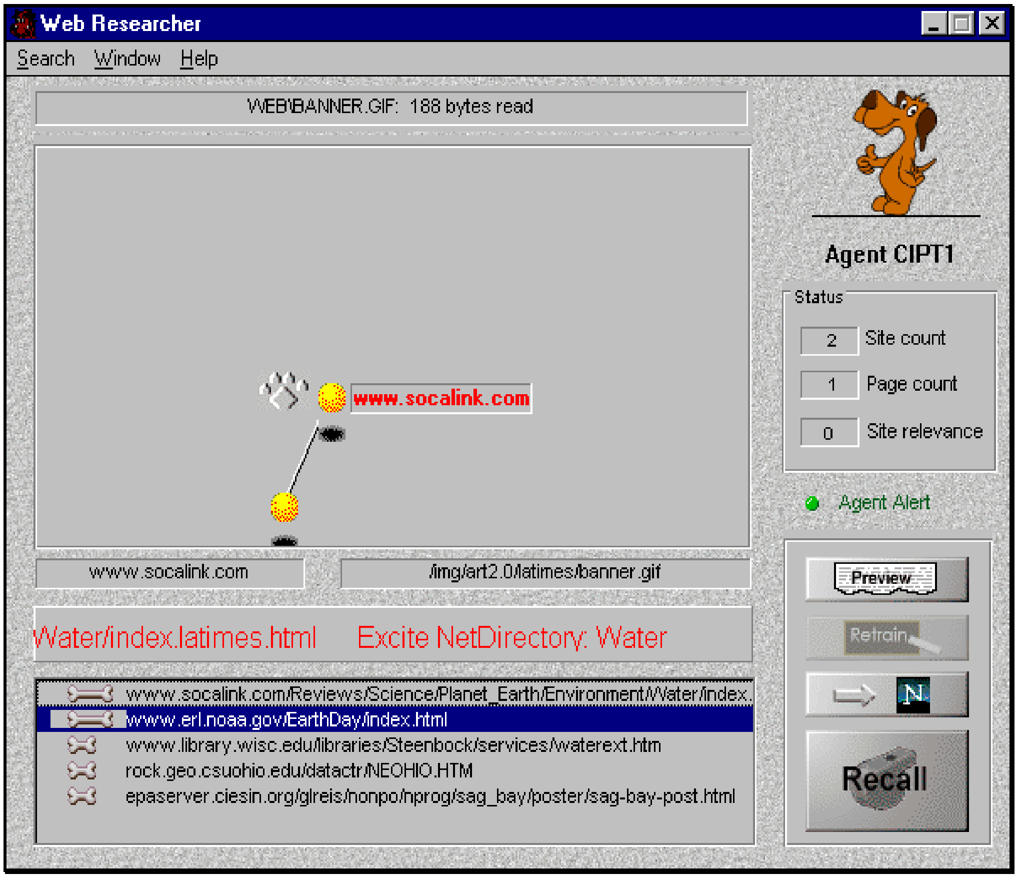
Figure 8.3: Autonomy agents ready for action



My mission is to find out
about environmental information
systems for pollution
in rivers, lakes, and coastal...

The next step is to try out your agent and progressively help it get better at finding what you want. You drag it onto the Web icon, and watch what happens (see Figure 8.4).

Figure 8.4: Autonomy at work



WEB\BANNER.GIF: 188 bytes read

2	Site count
1	Page count
0	Site relevance

www.socalink.com /img/art2.0/latimes/banner.gif

Water/index.latimes.html Excite NetDirectory: Water

www.socalink.com/Reviews/Science/Planet_Earth/Environment/Water/index.
www.erl.noaa.gov/EarthDay/index.html
www.library.wisc.edu/libraries/Steenbock/services/waterext.htm
rock.geo.csuohio.edu/datactr/NEOHIO.HTM
epaserver.ciesin.org/glreis/nonpo/nprog/sag_bay/poster/sag-bay-post.html

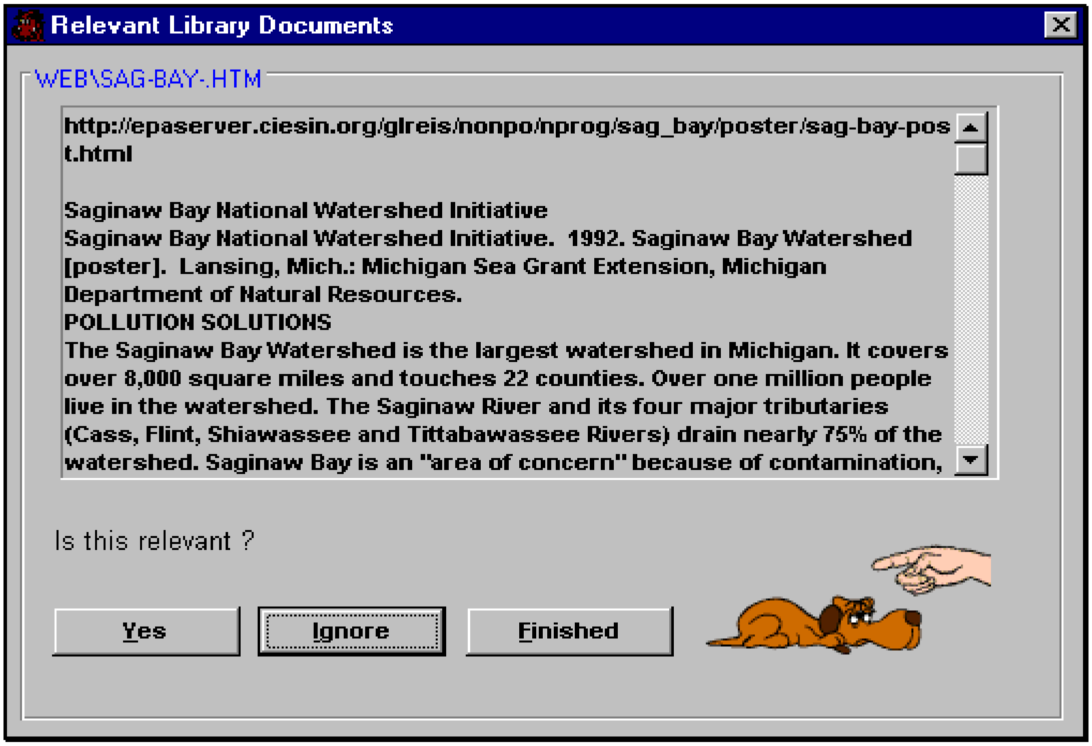
The little hound is highly animated at this stage, scratching, sniffing, and occasionally giving the ‘thumbs-up’ signal to indicate that it has found something useful. As it works, it brings the useful information back to your computer, ready for storing in the Library (see Figure 8.3). As Figure 8.4 shows, a lot of dynamic information is offered as the agent works: a map of where it is and has been, scrolling text highlighting the information currently being retrieved, and a preview panel in which documents can be seen as they arrive.

Autonomy uses a combination of neural network and fuzzy reasoning. The neural network identifies key concepts in the text. Autonomy then applies fuzzy reasoning, based on a stored language knowledge base, to decide the relative importance of the different terms. This allows it to identify other ideas that are related to the training text, but are not explicitly mentioned there. As it searches, Autonomy compares the pattern of each document to the pattern in the training you gave it. This means that unlike a conventional web search, Autonomy does not use keywords, but actually identifies the concepts involved in the text. Autonomy assigns a relevance to documents using the size of the bone as an indicator. After a few documents have been retrieved, you go through them with the agent (see Figure 8.5), chastising it for bringing back irrelevant ones, and rewarding it for useful information. This is the way the agent is trained. It is important to stay in control of your agents, particularly in the early stages of training, as they have a tendency to wander off in info-space.

Getting an agent to be really useful is not as easy as it sounds; it takes some practice (and luck!) to identify the right combination of initial mission statement and example data that will make your agent satisfyingly selective. Nevertheless, in our opinion Autonomy Web is a useful and far more engaging way of gathering information than conventional Web searches.

The real power of Autonomy comes when the Library is populated with the significant documents on your topic of interest. You populate the library by accepting documents found by different agents. When you want to work on information gathered by a particular agent, you drag that agent onto the library icon, and only documents retrieved by it are displayed. You can place a query on the library by typing a sentence or phrase, in the same way that agents are trained. Searching this local database is of course much faster than retrieving information across the net.

Figure 8.5: Training the agent



http://epaserver.ciesin.org/glreis/nonpo/nprog/sag_bay/poster/sag-bay-post.html

Saginaw Bay National Watershed Initiative
Saginaw Bay National Watershed Initiative. 1992. Saginaw Bay Watershed [poster]. Lansing, Mich.: Michigan Sea Grant Extension, Michigan Department of Natural Resources.
POLLUTION SOLUTIONS
The Saginaw Bay Watershed is the largest watershed in Michigan. It covers over 8,000 square miles and touches 22 counties. Over one million people live in the watershed. The Saginaw River and its four major tributaries (Cass, Flint, Shiawassee and Tittabawassee Rivers) drain nearly 75% of the watershed. Saginaw Bay is an "area of concern" because of contamination,

Is this relevant ?

System Requirements
Autonomy is available from Autonomy Corporation *(http://www.agentware.com)*. It runs on PCs under Windows 3.11 or 95. It requires an Internet connection, which can be either by modem or fixed line. It also needs a Web Browser, either Netscape (2.0 or higher) or Internet Explorer (3.0 or higher). Other system requirements are:

- at least 8 Mb of RAM (16 Mb recommended)
- 256 colour SVGA display or better
- enough free hard disk space to satisfy your appetite for information (20 Mb recommended).

Table 8.1 shows the cost factor rankings for Autonomy, using the categories defined in Table 2.1.

Table 8.1: Autonomy cost factors

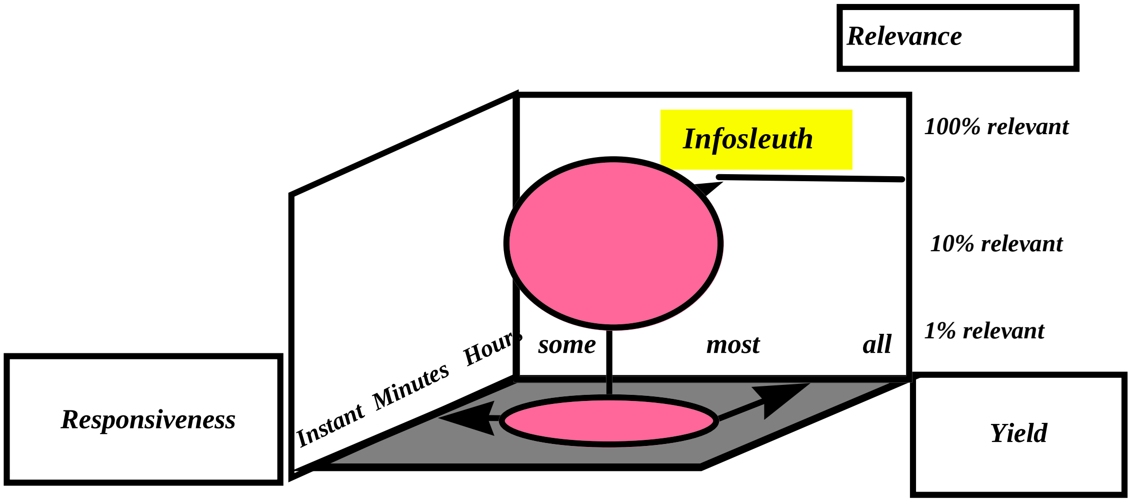
learning	installation	application	price
B-C hours to days	A standard windows installation	B-C hours to days	A

8.2.2. Infosleuth™

InfoSleuth™ is a research project run by Microelectronics and Computer Technology Corporation (MCC) in Austin, Texas *(http://www.mcc.com/projects/infosleuth/)*. The EEA together with three US agencies (EPA – environment, DOE – energy and DOD – defence) are collaborating in an Environmental Data Exchange Network (EDEN) project to develop and demonstrate of an improved means for sharing environmental data, based on Infosleuth technology[32] . See also: *http://www.mcc.com/env/EIMall_ToC.html.*

As the technology has not yet been proven, the diagram for Infosleuth can only be an opinion about what it might achieve if the project is completed successfully. Reflecting this, the outer ovals on Figure 8.6 show a wide range of possibilities and therefore the approach we will take in describing the capabilities of Infosleuth is in the wider context of developments in agent technology, rather than attempting to describe the characteristics of the tool in action.

Figure 8.6: Infosleuth™ selection ratings



The problems faced by the four agencies collaborating on the EDEN project is that environmental workers need to access multiple information sources, that may be held on different hardware and software systems and may have been stored in different ways. As with the WWW, there cannot be centralised control of the information sources, and thus information may appear and disappear without notice.

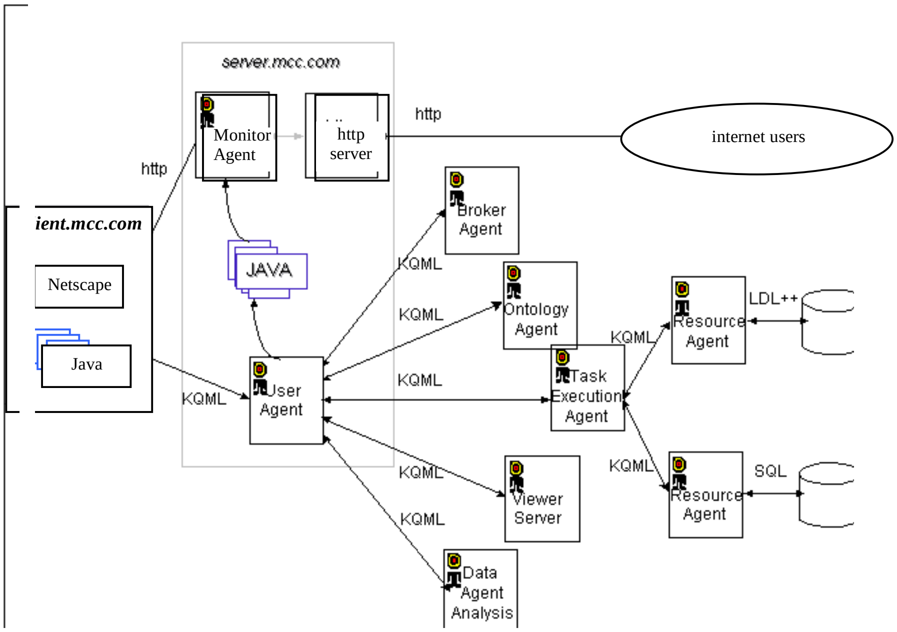
The InfoSleuth solution is a community of agents that co-operate to help the user find and retrieve needed information. The idea is that since it is difficult to make individual software components intelligent enough to accomplish complex tasks on their own, the approach should be to make simple units that can co-operate with one another to solve larger problems. The co-operative result (it is hoped) will be much more than the sum of the individual agents' capacities. To set Infosleuth in context of the wider agent research community, here is a sample of the issues that are currently being addressed by researchers:

- agent mobility – agents that are able to travel across the networks to accomplish their tasks
- co-operative planning – how do communities of agents decide what to do?
- user interface and control – how do humans retain the right degree of control of their agents?
- agent communication languages
- open agent architectures.

For the Infosleuth designers, the key issue was agreement among agents on the meanings of terms, i.e. ontologies (see buzzwords at beginning of chapter). To help solve this problem, Infosleuth uses an 'ontology agent' to store, retrieve, and manipulate different ontologies. Figure 8-7 shows the Infosleuth architecture.

[32] Much of the material in this section is adapted or copied from MCC documents, used with permission.

Figure 8.7: InfoSleuth architecture



Its main features are:

- **User Agent:** an intelligent gateway into InfoSleuth. It uses knowledge of stored ontologies to assist the user in formulating queries and displaying results.
- **Ontology Agent**: provides an overall knowledge of ontologies and answers queries about ontologies.
- **Broker Agent**: receives and stores advertisements from all InfoSleuth agents on their capabilities. Based on this information, it responds to queries from agents as to where to route their specific requests.
- **Resource Agent**: provides a mapping from a common ontology to the database schema and language native to its resource, and executes the requests specific to that resource, including continuous queries and notifications.
- **Task Execution Agent**: co-ordinates the execution of high-level information-gathering sub-tasks (scenarios) necessary to fulfil the queries. It uses information supplied by the Broker Agent to identify the resources that have the requested information, routes requests to the appropriate Resource Agents, and reassembles the results.
- **Monitor Agent**: tracks the agent interactions and the task execution steps. It also provides a visual interface to display the execution.
- **Data Analysis Agent**: corresponds to resource agents specialised for data analysis/mining methods.

The Crisp AI language 'KQML' is used to express queries and results and the InfoSleuth agents will be implemented, principally, in Java, as a means to achieve cross-platform compatibility and ready use on the Internet. Although these agents will communicate with each other over the network to request information and assign tasks, the complexity of this communication is concealed from a user by means of a User Agent that orchestrates all the necessary activities to answer the user's queries. An analogy is possible between InfoSleuth and a library. A 'patron' (user) of the library (the InfoSleuth system of networked resources) who wishes to learn about a subject might send an assistant (the User Agent) to the library to perform the necessary research. The assistant approaches the Information Desk (Broker Agent) to determine where to go. Based on the general topic of the query, the clerk directs the assistant to a research librarian (Task Execution Agent) who is familiar with the topic (the domain-specific ontology – in the library, the topics are defined in standard ways, e.g. Library of Congress, Dewey Decimal, or MeSH; in InfoSleuth, ontologies are defined in terms of the Environmental Data Registry or other domains). The research librarian consults the catalogue (Ontology Agent) for more specific information on the topic, and then the resource locator (a second function of the Broker Agent) for detailed information on where to look. The book/information is retrieved from the appropriate part of the stacks (requests issued to the appropriate Resource Agents) and the assistant, collates and prepares the specific information (selects an appropriate viewer Applet) for the patron's use.

This analogy can help explain the major difference between InfoSleuth and full-text indices of the WWW. Where a research librarian (Task Execution Agent) relies on the use of a catalogue organised by subject headings (the Ontology), the use of full-text indexing is best compared to looking through a massive dictionary where each word is followed by a list of all of the page references of all books that contain those keywords, independent of content.

8.2.3. Other Products

Table 8.2 provides summaries of a selection of agent packages – at least one example of the major approaches.

8.3. Environmental Examples

Unlike this section in the other chapters, here we describe how agent technology might be deployed in the service of the environmental community in the future. The technical capability to realise this scenario is already present[33]; it's just waiting for someone to take up the challenge of doing it. Policy makers need access to high-quality timely information. They must make decisions considering four kinds of information:

- facts
- recommendations for action provided by agencies
- public opinion
- the will of others taking part in the policy-making process.

Increasing pressures on the environment and society as a whole will mean that the pressure on policy makers to make correct decisions quickly will increase. In particular, there will be increasing demand for environmental agencies to examine and evaluate multiple decision options, often on very short timescales. This is because decisions are increasingly interconnected – economics affects environment, environment affects quality of life, economics and quality of life affects crime, etc.. A major impediment to rapid option evaluation is the lack of an appropriate central authority and the resulting need for inter-agency co-operation. Nevertheless, it seems clear that the capability to respond rapidly and effectively to proposals for change will be a key weapon in the battle to protect the environment. Agent-based technology may well play a role in establishing this capability. Box 7.1 describes one vision of how the notion of environmental agents might develop.

[33] The individual components needed are present in Infosleuth, for example, which should be capable of handling the problem. However, implementing such a scenario would require considerable resources.

Box 7.1 Responding to sea level rise predictions

Imagine a network of software agents, where a thematic agent, for example for water quality, is in touch with regional agents, each of which will be in touch with local agents. Other agents will represent the populace of regions, soil, various modes of transport, the atmosphere, even the sun! Leaf-level agents (those at the bottom of the hierarchy) will be in touch with the environment itself, through sensors and communication systems. Higher-level agents will be endowed with the capability to respond to pressures. The agents will be geographically distributed, being located at centres of expertise where they can be kept up-to-date and continuously on-line. These environmental agents will have several interesting properties:

- history: they will remember what happened in the past;
- functional capability: they will be able to respond to commands to compute things using their data base (for example statistical measures of state, and predictions of effect of new pressures);
- probabilistic capability: they will be able to respond with ranges of possibilities.

When an environmental agency is tasked with compiling a state of the environment report, commands will go out to all of the top-level agents, which will direct the others to compile the desired statistics. When a new measure is being considered, the relevant information will be passed down to the agents as a request for evaluation, and the network will be directed to produce relevant summary statistics. This might involve simulations of possible outcomes (see Chapter 4 – Optimisation for details), so that decision-makers can evaluate not only the range of possible outcomes, but also the probability of each.

Let us suppose that predictions of global change have advanced considerably, and forecasts of sea-level rise now range up to 10 metres over the next three decades. One option being considered is a system of dykes and pumping stations. Imagine that the direct engineering costs of the endeavour have been estimated, and now environmental agencies are tasked with evaluating the impact of the proposed construction on river systems, transportation networks, fishing, and agriculture. For simplicity, we ignore the weather aspects of the problem. The inputs to the problem are:

a set of future time series of sea-level, with probability values for each series;
a suite of engineering plans, suitable for different rise values.

For all the plausible combinations of sea level time series and dyke engineering plans, the network of environmental agents is tasked to simulate the response of different species and resources, again as a function of time. For example the one River Basin Agent might predict changes like the following:

- reduced catchment under some dyke engineering plans
- increased salinity of river water
- stagnation of smaller unpumped water bodies.

Agriculture and fishing agents connected to the River Basin Agent would evaluate the changes that would occur to their domains as a consequence of these and other changes; and so on through the complete network. Thousands of scenario simulations would be run. The final result would be risk-cost curves on the various options for the policy makers to evaluate.

Table 8.2: Other intelligent agent products

Product	Overview	Product features	Data limits	Platform / browsers	Ease of use	Comments
FreeLoader™	An off-line or on-line automatic surfer, which can download web sites for speedier surfing.	Can check sites at specified intervals and depth. Automatically notifies user when selected pages are updated.	10 Mb plus download space, 8 Mb RAM.	Windows 3.1 and 95. For Netscape only.	Easy to use graphical interface.	From FreeLoader Inc (http://www.freeloader.net). Available free.
Metabot™	A Java applet which simultaneously searches major search engines and returns a ranked list of URLs with the duplicates removed.	The search results are listed with the duplicates removed. After the initial search a 'webcrawler' can be deployed which visits every page of the search links and extracts the information of interest.	Java must be installed on the webserver. Java runtime environment can be downloaded free from Javasoft (http://www.javasoft.com). The server must have 4 Mb of free memory.	Runs on any computer with a web browser that supports Java.	A wizard window helps create the webcrawler. Takes longer to install and use as it lacks the familiar Windows environment.	From Kinetoscope Inc (http://metabot. kinetoscope.com).
Surfbot™	For off-line and on-line browsing and searching. Available as a plug-in.	Actively fetches, filters and organises information to user specification, monitors bookmarks, runs schedule searches and map sites. Works on its own or as the client component of the Surflogic Agent System. Delete unwanted URLs to refine the Agent.	On-line time and cache size limits can be selected.	Windows 95 or NT 4.0. Netscape or Navigator, Version 3.0. Compact installation (<500 K download).	On-demand interface from an in-line browser plug-in. Single-click and drag-and-drop features. Options to 'stay on site, stay in folder and go outside site'. Select folders, locations, bookmarks or keywords. Documentation is available on-line.	From SurfLogic LCC (http://www.surflogic. com).
Trudger™	A multi-platform web agent which does not require a web browser to perform its function but can communicate with one to display the downloaded pages.	The user defines a suitable starting point. Monitors sites, indicates sites to avoid, number of links to transverse and set time and disk space limits and a schedule. Interrupted downloads can be resumed later and the number of parallel downloads, specified. Searches redirected by pausing an in progress download and deleting unwanted pages.	Three configurations: TrudgerLite, Trudger Professional-Home/Personal Use, Trudger Professional-Commercial Use. Can be upgraded.	Windows 95 and NT, Unixware, Sun Solaris/SunOS (Sparc, Intel), HP/UX, IBM RS6000 AIX, SGI, SCO, BSD, Linux, DEC-Alpha, LynxOS platforms.	It is up to the user to specify a good starting point. A lot of fine-tuning and knowledge of which types of file are not needed is required to download only relevant data.	From Vital (http://www.vital.com).
Web Retriever™	Off-line web browser and information retrieval product. Searches links for specified data at designated time.	Compresses and organises data into an infobase. Allows text editing, searchable highlighters, searchable note annotations, bookmarks and groups. Automatically saves retrieved information.	3.5 Mb plus download space, 8 Mb RAM.	Windows 3.1, 95 and NT 3.5 (or later) and Macintosh. Works with all browsers.	The user interface is similar to that of the browser. Makes web content portable.	From Folio Corp (http://www.folio.com).
WebCompass ™	An intelligent searching agent. Input areas of interest and the program automatically finds sites that most closely match the criteria.	Searches up to 70 search engines. Schedules searches and monitors web sites. Ranks and summarises information. Organises results by topic. Enhances search by linking and adding topics.	486 or Pentium compatible processor. 8 Mb RAM and 5 Mb hard disk space.	Windows 95 or NT (3.51 or better).	Has a basic and advanced on-line tutorial. Help Wizard to provide guidance.	From QuarterDeck (http://www.quarterdeck.com).

Product	Overview	Product features	Data limits	Platform / browsers	Ease of use	Comments
WebDoggie™	Personalised web document filtering system using collaborative filtering.	Users evaluate web pages and WebDoggie learns from the preferences and from other registered users with similar interests.		A client-server system. Runs on various machines on the Internet. Use the WWW interface or install the WebDoggie client and modified XMosaic-2.4 browser on your machine.	Simply rank pages on a scale of 1-7 or use keywords.	From MIT Media Laboratories (*http://webhound.www.media.mit.edu/projects/webhound/*).
WebEx™ (formerly Milktruck Delivery)	Browser enhancement automatically downloads web sites for later surfing and notifies user when selected pages are changed.	Maintains presentation, links, images and sound from web site. Allows scheduling of information retrieval.	2 Mb plus download space, 8 Mb RAM.	Windows 3.1, 95 and Windows NT. Netscape and Internet Explorer	Integrates with browser to eliminate the learning of a new interface	Travelling Software, Inc. (*http://www.travsoft.com*).
WebFetcher™	Downloads web pages to hard disk for off-line viewing.	A list of URLs can be scheduled for download to a certain depth. The sites are periodically checked for new or updated information.		Available for Macintosh 7.0.1 or later. Windows 95 and NT. UNIX versions NeXT, SunOS, Solaris 2.3, OSF/1 for Dec Alpha, Dec Ultrix 4.3 and Linux with a live Internet connection.	The interface is not graphical so there is less pointing-and-clicking. Some documentation is available on-line	From OnTV (*http://www.ontv.com*)
WebInterests! ™	Enables web servers to notify users of new information and changes to web sites of interest.	Filters documents to user specifications, documents processed on-the-fly. Receives active notification, customises results and design templates. Summarises documents over a specific size.		Windows NT and UNIX; SunOS, Solaris, HP-UX, IBM AIX	Templates for the presentation of results need to be designed which may not be a task for novices.	From InTEXT Systems (*http://www.intext.com*).
WebSeeker™	Retrieves search results and monitors specified web sites.	Combines 23 search engines, eliminates duplicate results, indexes and refines results, schedules unattended searches and monitors web pages for recent changes.		Windows 95 and NT.	Comes with on-line help and a Quick Tips file. Customer support available by telephone, e-mail and FAQs.	From ForeFront (*http://www.ffg.com*).
WebWhacker ™	For off-line browsing	Schedules downloads, monitors sites for new information and updates.	10 Mb plus download space, 4 Mb RAM.	Windows 3.1, 95 and NT and MacOS. For Netscape only.	Has a toolbar to select and retrieve web pages without leaving the browser. Help Wizards guide users through all operations.	From Forefront (*http://www.ffg.com*) Available in English, German and French.
WiseWire™	Allows collaborative filtering of information and uses user feedback to retrieve only relevant information. Based on neural networks.	WiseWire 'learns' user preferences to filter information. Preferences of users with similar interests are taken into account. User interests can be grouped into personal magazines and themes. A premium service which includes greater customisation and features		Available for any PC running a web browser.	As you view web pages in WiseWire, you give each page a relevancy rating. After just a small number of articles, more relevant information is retrieved.	From Empirical Media Corp (*http://www.wisewire.com/emc/*). Basic service available free.

8.4. References and Bibliography

Bradshaw, J. (ed). 1997. Software Agents, MIT Press.

Herman, B. 1996. Intelligent Software Agents on the Internet: – an inventory of currently offered functionality in the information society and a prediction of (near-)future development. End Thesis, Tilburg University, Tilberg, The Netherlands. *(url ref: http://www.hermans.org/agents)*

Mejer, R. 1996. Intelligent software agents: perspectives for business. In: The IPTS Report, No. 5. Joint Research Centre publication GK-AA-96-005-EN-C.

Rasmussen, L. 1997. Using agents to secure the internet marketplace reactive security and social control. Proceedings of the Second International Conference on the Practical Application of Intelligent Agent and Multi-Agent Technology. The Practical Applications Company.

GLOSSARY

Agent : Software units that can act on their own initiative on behalf of the user, based on general knowledge about the user's goals and preferences

Algorithm : A set of ordered steps for solving a problem, such as a mathematical formula or the instructions in a program. Examples: numerical integration, weather prediction, evaluating a neural network.

Alphanumeric : Containing only alphabetical and numerical characters.

Anomalous : Difficult to understand in the light of other information.

Architecture : The principal concepts on which a computer program is developed. Just as buildings should be designed before they are constructed, so should computer programs. This is not always done.

Artificial Intelligence : A style of programming computers by stating rules and facts, rather than describing in detail how the computer should perform.

Backpercolation : A special training algorithm for multi layer perceptrons

Back Propagation :-A widely used algorithm for training multi-layer perceptrons, also used as a synonym for this neural network type. 'Back-prop' for short.

Bayesian Expert System :-Expert system technology based on Bayes' theorem, which relates conditional and prior probabilities; i.e. the chance that something will occur based on direct evidence and the chance that something will occur based on its likelihood of happening at all.

Binary Logic : Reasoning with ones and zeros. Elements of the system are either true or false. See also crisp artificial intelligence.

Bootstrap Sampling : A very general way of determining distributions or parameters estimated from measurement data. Bootstrap sampling assumes that all the information about the distribution is contained in the measurements data. Given a uniform set of measurements {S} of size N, from which a parameter P is derived, obtain the distribution of P by drawing many random samples from {S}, each of size N. From each sample, P is estimated, and the histogram of P over all the samples is plotted.

Broker : An agent that helps other agents, by providing connections, translations, or other similar services.

Classifier : A program that sorts data into categories. For example classifying the pixels of a remote sensing scene of an agricultural area according to the crop type grown.

Code : Program.

Compiler :-Computer utility to translate programs that have been written in a convenient language for programmers into low-level instructions the machine can run.

Confidence : The strength of belief that something is true, usually expressed as a percentage, where 100% = complete confidence.

Convergence : Gradual coming together or approach to a steady state. Environmental damage may converge to an acceptable steady state, eventually. Computer algorithms that find solutions iteratively will converge, if they are successful.

Correlation : The degree to which one variable is related to another. Expressed as a number in the range 0 to 1.

Correlation Plot : Graph showing one variable versus one or more others to identify relationships, trends, and classes.

Covariance Structure Analysis : A way of analysing how variables in a dataset relate to one another.

Crisp AI : Reasoning programs that use crisp sets. Each reasoning element returns a definite true or false result.

Crisp Sets :-Also known as conventional sets, boolean sets, or just sets. Billiard ball colours are a crisp set :- each ball has exactly one colour. Human skin colours do not form a crisp set.

Data Mining : Searching in databases for undiscovered relationships.

Deterministic System : Having a definite outcome, as opposed to probabilistic or fuzzy.

Enumerated Variables : Variables that can take only one of a set of values, such as {true, false}, or {red, green, blue}.

Error Propagation : From the errors in the inputs of a system, evaluation of the errors in the outputs.

Extrapolation : Estimating a value which is outside the range of a reference dataset.

Fuzzy Algebra : Logic rules for fuzzy sets.

Fuzzy Sets : Sets in which members may have partial membership, for example, the set {children, adults} in which teenagers belong to some degree to both. Environmental incidents might be classed into the sets {minor, major and catastrophic}, but the complexity of events will probably mean that there are aspects of all of these categories for some incidents.

Fuzzy Expert System : Expert system built on the principles of fuzzy sets and fuzzy algebra.

Fuzzy System : A system in which the outcome is vague and thus expressible in terms of fuzzy sets.

Gaussian Distribution : A random distribution that has a characteristic bell-shape. Also known as a 'normal' distribution.

Histogramming : Plotting the frequency with which data falls into defined categories to reveal the characteristics of the underlying distribution.

Hypertext:A computer-based document containing links between related text. For example, by selecting a word in a sentence, information about that word is retrieved if it exists, or the next occurrence of the word is found. Hypertext is the foundation of the World Wide Web (see WWW). Links embedded within Web pages are addresses to other Web pages either stored locally or in a Web server anywhere in the world.

Genetic Algorithm :- A general problem-solving technique based on ideas borrowed from evolutionary theory. A special kind of Monte-Carlo method.

Infosleuth : A large international agent-based environmental information research project.

Interpolation : Estimating a value inside the range of a reference dataset.

Iteration : Computer jargon for repeatedly doing the same thing, for example the act of testing a number of water samples could be described as iterating the test over the samples. Programs often find an approximate solution, then iterate their calculations to improve the answer.

Kinaesthetic : Involving the body sense of balance, motion and posture.

Linear Programming :-A collection of matrix-based techniques for optimising first-order (linear) quantities that is applicable when the relationships among the variables are linear equalities or inequalities.

Media : Materials that hold data in any form or that allow data to pass through them, including paper, transparencies, hard disk, floppy disks and optical disks, magnetic tape, wire, cable and fibre. Media is the plural of 'medium'.

Metadata : Information about data, such as when and where the data were collected and where it is stored.

Model : A mathematical/computer description of a system that can be used to predict the systems behaviour. Examples range from simple equations to giant numerical weather prediction codes.

Monte-Carlo Methods :-Simulation technique useful in a wide variety of problems where the complexity of the problem makes analytical solution impractical or too costly. Builds up situation statistics by choosing random values for variables, simulating an outcome and repeating this a great many times.

Multi-Layer Perceptron :-A variety of neural network in which there are three or more layers, each layer consisting of a number of nodes. The calculations of each node in each layer depend on the values of all the nodes in the previous layer, which is called 'fully connected'. Also known as 'back-propagation' or 'back-prop'.

Neural Network : A general class of computer programs loosely based on the organisation of biological neurons. Used in applications such as robotics, diagnosing, forecasting, image processing and pattern recognition.

Neurofuzzy : Technology that combines aspects of fuzzy systems and neural networks.

Noisy Data : Data in which there is a high degree of random variation masking the effect of interest.

Non-Linear Optimisation :-See linear programming. A collection of techniques that apply when the relationships are non-linear.

Object-Oriented :-A style of computer program design that enhances flexibility and extensibility by using the features of real-world objects as an organising principle.

Ontology : The set of terms that apply in some domain.

Optimisation : Adjusting the parameters of a system so that its outputs are best in some sense. A trivial example is adjusting the control on a video screen for best viewing. A more complex example is adjusting the controls in a water treatment plant to achieve acceptable water quality at minimum cost.

Outlier : Data point laying far from the main distribution.

Package : Synonym for software.

Parameter : A factor in an equation that is estimated from data. For example in a straight line equation y = ax + b, where y and x are data, a and b are parameters. See variable.

Pixel : A single vision unit on a display screen or image. Computer screens typically consist of 600 rows of 800 pixels each.

Platform : A particular kind of computer and operating system. Examples, unix, or a PC running Windows 95.

Probabilistic System : Having an outcome subject to chance.

Production Rule :-A statement in rule-based programming. Consists of an IF clause and a THEN clause. IF (waste is hazardous) THEN (use safety precautions for removal).

Program : A set of instructions for a computer.

Random Access Memory :-Electronic storage of a computer. Volatile, in contrast with non-volatile media such as disk or CD-ROM.

Regulation : Laws or rules that govern acceptable practice, for example emission regulations.

Risk Management : The process of identifying and quantifying the uncontrolled elements in a situation and devising measures to manage their impact on the whole situation. Examples : flood control and project management.

Rule-Based Programming : Synonymous with AI, particularly crisp AI.

Rule Confidence : A concept in fuzzy expert systems, in which a rule may apply to some degree. Rule confidence is expressed as a number between 0 and 1.

Scatter Plot : A two-variable correlation plot.

Scriptable : Allows the user to define a series of instructions that can be executed as a group on command.

Simulation Data : Data generated by a computer program designed to simulate the behaviour of some system. Example : weather models.

Software : Program and accompanying documentation.

System : A group of factors that interact in some defined way. For example weather systems, computer programs, ecosystems.

Uni-variate Relationship :-Depending on only one variable. Example :- The distance a body falls depends only on time (neglecting air resistance). Counter-example :- the distance a bird can fly depends on its weight, wing size, and other variables. This is a multivariate relationship.

Validation : The activity of verifying that a model is correct.

Variable : A factor that varies naturally and influences the result of an equation. See parameter.

Volatile : May vanish. For example, all the information in RAM is lost when the computer is turned off or crashes.

World Wide Web : A global information network based on http and html which allows users to find and view text and graphical information provided by others.

ACRONYMS

ADL	Application Development Library
AI	Artificial Intelligence
ANN	Artificial Neural Network
BAI	Best Available Information
BMP	Bit MaP file format
CERN	European Laboratory for Particle Physics
CIPTs	Computer Intelligent Processing Technologies
CLIPs	C Language Integrated Production System
CYC	from enCYClopedia, a giant expert system project
DLL	Dynamically Linked Library
DoD	Department of Defense (USA)
DoE	Department of Energy (USA)
DOS	Disk Operating System
EC	European Community
ecu	European Currency Unit
EEA	European Environment Agency
EPA	Environment Protection Agency
EUFIT	European Congress on Intelligent Techniques and Soft Computing
EURIDIT	European Network in Uncertainty Techniques Developments for Use in Information Technology
FAQs	Frequently Asked Questions
FES	Fuzzy Expert System
FFT	Fast Fourier Transform
GAs	Genetic Algorithms
GIF	Graphics Interchange Format
GIGO	‘Garbage In, Garbage Out’
GIS	Geographic Information Systems
GUI	Graphical User Interface
HPÔ	Hewlett Packard
html	Hypertext markup language
http	Hypertext transfer protocol
IBMÔ	International Business Machines
IDLÔ	Interactive Data Language
IEEE	Institute of Electrical and Electronic Engineers
JPEG	Joint Photographics Expert Group file format
Kecu	thousands of ecu
KQML	Knowledge Query Manipulation Language
LP	Linear Programming
MB	megabyte
MCC	Microelectronics & Computer Technology Corporation
MLP	Multi-Layer Perception
MPEG	The Moving Picture Experts Group
na	not applicable
NASA	National Aeronautics and Space Agency
NN	Neural Network
ODBC	Open Database Connectivity
ORM	Optimisation and Risk Management
PICT	PICTure file format
RAM	Random Access Memory
RBF	Radial Basis Function
RDBMS	Relational Database Management System
SQL	Structured Query Language
TIFF	Tagged Image File Format
URL	Uniform Resource Locator
WWW	World-Wide-Web