KFRI Research Report 118

EXPERT SYSTEM FOR DESIGNING EXPERIMENTS IN FORESTRY

P. Rugmini
K. Jayaraman
V.K. Bhadran

KERALA FOREST RESEARCH INSTITUTE
PEECHI, THRISSUR

June 1997 Pages: 24

CONTENTS

ABSTRACT

The possibility of using expert systems in designing experiments in forestry was explored by organising the concerned knowledge base and incorporating the same into an expert system shell. Designs commonly used in forestry experiments in different contexts were first identified and the circumstances under which these designs can be used were worked out. The algorithms for constructing the designs formed the knowledge base for the expert system. The knowledge base incorporated into the expert system shell GENEX formed the product designated as DESIGNEX.

The software DESIGNEX runs under DOS environment. Minimum memory of 640 is required for running the software. The system can identify appropriate experimental designs based on the user-given information about the experimental context. The programme also generates field layout plans of the designs for the user. The product will be useful to researchers in forestry and allied fields in designing their experiments.

1. INTRODUCTION

Computers are nowadays used in almost all fields of human endeavour. However, most of the present day applications have so far utilised only the enormous memory and computing ability of the computers. Efforts were on way to inculcate capabilities such as creativity or problem solving abilities into the working mode of these machines which resulted in the fast growing field of Artificial Intelligence(A1) systems. One of the most successful A1 technologies is that of expert systems(ES), which are intelligent computer programmes for solving difficult problems at a level comparable to a human expert.

The tasks of expert systems have been classified into six categories, viz., diagnosis, monitoring, prediction, planning, designing and interpretation (Kourtz, 1990). Diagnostic systems are the most successful expert systems which can be used in the field of medicines and electronics. Diagnostic systems can be created in the field of forestry to handle problems such as dealing with forest pests and diseases, 'interpretation of site classification in silviculture and genetic mapping. Monitoring systems are currently used in medical, nuclear and chemical industries. Forest fire management, pollution monitoring, insect pest management as well as forest nursery management are some of the promising areas where such systems are required. Prediction systems are presently used in meteorology, stock market analysis and watershed management. Forest growth modelling, insect dynamics and wood supply and demand projection are potentially prospective areas for forestry prediction systems. Many large planning systems, such as naval fleet management which are being developed, are useful to insure consistent knowledge and policy applications. Outbreak of insect pests, forest fire control resources, silviculture and harvest scheduling are examples of such forestry systems. Forestry application of design systems include forest inventory layouts, regeneration survey design and experimental design aids. Interpretation systems are used in mass spectrometry for chemical analysis, in speech analysis, in geologic explanation and in robotic vision. In forestry, interpretative expert systems can be developed in forest genetics, for interpreting remote sensing images mainly for temporal changes and with geographic information systems to convert them into geographic knowledge bases. Major applications of the expert systems are found useful in natural resource management which include pest and disease control, forest fire management and remote sensing (Kourtz,1990 and Latin *et al.*, 1990).

Jayaraman and Rugmini (1990) identified an application of expert system in choosing appropriate statistical techniques needed for various scientific studies in forestry, which is a case of design systems as mentioned above. Such a statistical expert system can be made to replace a statistical consultant. The knowledge of a statistician once stored in a computer can be accessed in an interactive manner with an expert system. The rulebase developed by Jayaraman and Rugmini(1990) was made operational through a package by name IITMRULE. This package had certain limitations which restricted the use of the system to merely identifying the statistical technique by the user. It was thought highly desirable to extend the system capability to execution phase of these techniques. Thus an expert system for designing experiments in forestry was conceived to be useful in choosing a design for scientific investigations and also in getting layout plans of the experimental design. For instance, if the system has suggested a complicated design like confounded or simplex centroid design for a given experiment, the user will be benefited by obtaining a field layout plan for the design. With this in mind, a project was formulated in collaboration with CIRA(Centre for Informatics Research and Advancement), ER & DC, Thiruvananthapuram, to develop an expert system for designing experiments in forestry. The expert system shell, GENEX developed by the latter has been made use of for this study. As the topic of expert systems and their applications is relatively new to forestry, an overview of expert systems in its many aspects is included in this report before describing the specific expert system developed during the course of this study.

2. EXPERT SYSTEMS : AN OVERVIEW

Expert systems permit us to encode human knowledge in the form of facts and rules related to a specific task or domain. The knowledge can then be used by the expert program to automatically analyze a situation and recommend solution as if a human expert were present. Expert system should be easily modifiable, flexible, incrementally buildable, capable of explaining its line of reasoning, able to learn by itself if possible, etc. In order to achieve these objectives we have to choose an appropriate framework for the expert system.

2.1. Organization of an expert system

Expert systems generally consist of an inference engine, a knowledge base, an explanation mechanism and a data base. The inference engine contains the general knowledge for executing strategies. The complexity of the inference engine hinges on the structure of the knowledge base and representation scheme used for knowledge base. The simplest inference engine just applies each chunk of knowledge in a knowledge base sequentially. It should also generate adequate information for the explanation mechanism in case explanation is asked by the user. The knowledge base incorporates the domain-specific knowledge organized in the form of independent chunks of knowledge which can be modified, deleted, or augmented. Thus the knowledge base is considered as an independent module just by augmenting the knowledge base without necessitating any modification to the inference engine. The data base contains data relevant to a particular problem. The explanation mechanism gets its input from the inference engine and then interacts with the knowledge base to put out the line of reasoning used or the sequence in which each chunk of knowledge has been used by the inference engine in arriving at the solution. It is expected to provide the explanation in a language as close to the natural language as possible.

2.1.1. Inference engine

Inferende engine is an interpreter-cum-scheduler that interprets the application of a rule on the current data or situation of the problem and decides which rule is to be fired next or which path is to be chosen next for proceeding with problem-solving. The universal strategy for generating alternatives and choosing from among these hinged on certain criteria which is usually employed in most expert system as the strategy for problem-solving. For most real life problems, the alternative solutions increase exponentially as the computation proceeds. Thus heuristic knowledge is resorted to for limiting the search. The major search strategies are depth-first search, breadth-first search, alpha-beta pruning, etc. In these processes, two types of reasoning are normally employed, viz., forward reasoning and backward reasoning.

Solving a problem means reaching a goal state starting from an initial state. This normally involves passing through a long chain of intermediate steps or states. When the program works from the initial state towards the goal state, the process is called forward reasoning. Forward reasoning is a good technique to use when all or most paths from any one of the initial states converge in one or a few goal states. An alternative to this is backward reasoning. In this case, the program begins to look for a path through the problem by starting with a goal state and seeing how it can be modified closer to an initial state. The situation in this case represents many goal states converging on one or few initial states. The goal state is converted to one or more sub-goals that are easier to solve and whose solutions are sufficient to solve the original problem.

Forward reasoning can be viewed as data-driven or bottom-up while backward reasoning is goal-directed or top-down. Typical data driven problems include process control, process monitoring, and some scheduling and planning problems. Backward reasoning schemes assume a solution and reason back to the data that support the assumption. They are with limited number of solutions where it is preferable to prove one particular solution than to analyze all the available data. Backward reasoning is useful for diagnosis and debugging problems.

Though much human problem solving behaviour is observed to involve reasoning backwards. it is often desirable to effectively integrate combination of forward and backward reasoning strategies for extensive real-world problem solving.

2.1.2. Knowledge base

Knowledge consists of facts and heuristics. Facts are knowledge that is agreed upon by the experts, publicly known, available in text book, etc. Heuristics are rules-of-thumb generated by an expert for his own use and are seldom publicised, but, are an important factor that determines the performance of an expert. Facts are proven-knowledge whereas heuristics are the basis for good judgements in the absence of deterministic knowledge.

The organization of knowledge base is important for its manipulation as well as for modifications. A highly modular approach provides easiness in updating the contents. Each knowledge chunk will be independent and have uniform structure. But, this results in increased overheads on storage space and search time. As the size of the knowledge base increases, search operations become time-critical. It will then be desirable *to* organize the knowledge chunks in a suitable hierarchy so that access to a particular portion can be well-guided and expedited. Basically, this involves classifying the information and relating different groups in a meaningful manner. But, introduction of groups and group hierarchies degrades the flexibility of knowledge base limiting the independence of its individual components. Thus the requirements of modularity and search efficiency are conflicting and a judicious choice of the levels of each determines the performance of the system.

2.1.3. Explanation mechanism

A human expert in addition to solving problems, is capable of explaining how he has arrived at the solutions thereby generating confidence in his solutions. This is an important aspect that makes his solutions acceptable to the users. In a similar fashion, the knowledge-based system should also be able to explain its solutions in sufficient details. Therefore incorporation of an explanation mechanism is an important aspect of knowledge based systems. Typically, the explanation mechanism gets its input from the inference engine and then interacts with the knowledge base to put out the line of reasoning used. The system must be able to answer questions like Why, Why not or How, etc. In addition to building more user faith in the system, explanations also help in debugging the knowledge base rules during the developmental phase.

2.1.4. Database

The data base is an organized collection of related information (data). The organized information or data base serves as a base from which desired information can be retrieved; many meaningful conclusions can be drawn, or decisions made, by further reorganizing or processing this data. Data base technology is now well developed. Most of the organizations which may be the probable users of expert system will have conventional data base with them. Expert system can be designed in such a way as to use the existing data base by providing appropriate data base calling functions.

2.2. Advantages of expert systems

The importance, interest, and enthusiasm exhibited for adopting this technology wherever possible stems from the following advantages.

(i) The total performance or capability of a software system depends on the quality and size of the knowledge that is used. In conventional programmes, the knowledge is embedded in the code and distributed throughout the program, thereby making it extremely difficult to assess and update the knowledge that is used. In an expert system, the knowledge used for problem solving is transparent and is isolated from the inferencing strategies. The transparent nature of the knowledge base enhances the capabilities and quality of the software. In other words, it permits incremental building up of the knowledge base and consequently the capabilities of the software package.

(ii) It can easily provide explanations for the conclusions it has arrived at. This helps in verifying the correctness of the knowledge base. It can also tell the informed user why it needs certain information at different stages of problem solving thereby increasing the confidence in the effectiveness of the system.

(iii) An expert system can combine the knowledge of many experts. Therefore, the expert system can perform at a level higher than any single human expert. Expert systems enable us to capture the knowledge of co-operating experts and use it for posterity.

(iv) This technology permits the development of applications in narrow domains in probably one-tenth the time required for developing equivalent systems using conventional programming techniques. In many cases, the uncertainties involved in the size of the knowledge base often makes it difficult to develop equivalent systems using conventional programming techniques.

2.3. Building of expert systems

Building an expert system from conception through integration takes a long time. The process involves transformation of knowledge in a discipline to a machine processable form. The first step is elicitation of knowledge from domain experts, followed by organizing this knowledge into some systematic representational fbrms. The process of capturing the knowledge of an expert and translating it into a form that could be used by the machine is known as 'knowledge engineering'. The knowledge engineer is a computer scientist who interacts extensively with the domain expert. Most often he has to have many discussions with the expert to elicit knowledge. In order to open a dialogue with the domain expert, the knowledge engineer should first familiarise himself with all the terminologies used in a particular domain, which is very time-consuming. Secondly, the expert may find it very difficult to explain many of the things to the knowledge engineer. This and all other related problems have inhibited the growth and the use of this technology in many fields.

Expert systems are usually built through expert system shells which are 'off-the-shelf expert systems with empty knowledge base, contain a predefined inference engine that knows how to use the knowledge base to arrive at a conclusion. In other words, a program used to develop an expert system is normally called an expert system shell. In the same way that a word processor is not a letter or document, but, a shell that helps the creation of text file, expert system shell helps in the creation of expert system. Expert system shell can be considered as a reasoning system out of which all the knowledge has been emptied. When the knowledge about a new domain is entered into a shell appropriately an expert system is created. The original shell can then be used to create a new expert system in a similar fashion.

An expert system shell performs the following major functions : It assists in building up the knowledge base by allowing the developer to insert knowledge into the knowledge representation structure. It provides a method of inference of deduction that reason on the basis of information in the knowledge base and new facts input by the user. It provides an inference that allows the user to set up by reasoning tasks and query system about its reasoning strategy. An expert system shell should be expected to have a production rule system, preferably capable of varying inference strategies. The ability to implement symbolic reasoning is an essential feature of an expert system shell. The expert system shell is designed for reasoning. One should not use a shell to carry out tasks, such as mathematical programming, that are performed well by numerical reasoning languages. Expert system shell improves productivity in building the expert system and also encourages non-computer professionals to transfer their expertise without extra efforts in learning computer shells.

3. GENEX - General purpose expert system shell

The tool which was used for developing expert system for designing experiments in forestry is GENEX, an expert system shell developed at CIRA (Centre for Informatics Research and Advancement), ER and DC, Thiruvananthapuram. It runs on MS-DOS environment. The language used for developing the shell is KLISP, a version of LISP developed at CIRA. GENEX is an expert system shell for general purpose applications in a wide range of domains such as design, diagnosis. planning etc. It supports various type of knowledge forms such as rules, relations, and procedural knowledge. A more detailed description of GENEX is available in Anonymous (1993). GENEX is still under development.

The knowledge base of the GENEX is a collection of functions and their descriptions. In addition to functions and their descriptions, the knowledge base also contains descriptions of global vocabulary (terms common to all functions), user-defined LISP functions and a description of the package.

Each function is described in terms of rules and relations. This describes the basic sequential statements that are executed or evaluated when the function is invoked. The rule or action is normally executed in the order in which they appear in the function list. The action commands typically include assignment statement, data manipulation commands input/output commands, etc. Rules have typical structure such as **if [condition] then [action] else [action].** Relationships between one variable with another set of variables are expressed typically as a set of equations.

Every function has a set of variables termed as inputs and outputs (to pass/collect values to/from other functions), a sequence of rules and action statements which is the main body of the function, and a set of relationship expressions illustrating the relation between different variables. As a number of terms will be used in expressing the rules and the relations, a provision to describe this vocabulary is also supported. This description consists of classifying the terms as variable, constant, proper noun. etc. and depending on their category, additional details are also to be supplied. The details of variable includes whether it is user-given, computed, or data base variable. prompt message for collecting the value, any explanations or help message, etc.

The user can translate the problem into rules in a natural way using his own terms without bothering about the type of the variable to be used and complete the entry of rules for the problem. During compilation, GENEX systematically collects various information from the user regarding the terms used and are stored in the knowledge base.

4. KNOWLEDGE BASE DEVELOPMENT

The success of an expert system depends on the way in which the knowledge base is prepared with anticipated response from the user. This can be achieved only by an expert in the field. Moreover, the expert should be able to interact with computers and computer specialists during the development. The knowledge base for designing experiments in forestry has to capture the expertise of the expert (statistician) and use it in deciding about the experimental design. Certain general remarks on the knowledge base with respect to the choice of designs are mentioned in the following.

Almost all experimental designs are based on the three most important principles viz., randomization, replication and local control. Randomization and replication are required to obtain a valid estimate of error. Blocking (local control) is essential to reduce the effect of heterogeneity of the experimental material on treatment comparison. With large number of treatments, the concept of blocking is extended to that of incomplete block designs. Multifactor experiments are needed to study interaction among several factors. Confounding is a means to reduce the block size in the case of factorial experiments, where as fractional factorial designs bring down the number of treatment combinations. Certain specialized designs like response surface designs, mixture designs or designs for bioassays are used for specific purposes or under certain constraints on the factor space.

Other than the above theoretical aspects, a number of practical considerations also affect the choice and layout of experimental designs. For instance, the number of replications to be used in a lattice design is mostly governed by the availability of the experimental material and accordingly the choice of the design diverts to simple lattice, triple lattice or balanced lattice. The possibility of missing values and associated complexities in analysis prevent many experimenters from choosing designs like partially incomplete block designs or confounded factorial experiments. Optimum plot size and shape, block size and shape, size of plot border, etc. are still to be worked out for many types of experimental materials. The user is prompted to use his/her discretion in such matters.

5. DESIGNEX - EXPERT SYSTEM FOR DESIGNING EXPERIMENTS IN FORESTRY

5.1 Scope and mode of operation

This project was aimed at developing an expert system for designing experiments in forestry. The product realized was designated as DESIGNEX which is described in detail in the following.

DESIGNEX was developed by using expert system shell, GENEX. Eventhough the shell includes a menu function key, labels and mouse cursor, none of these are used in this programme for running purpose. The system encompasses most of the commonly used experimental designs in forestry. The designs specifically covered by the DESIGNEX are discernible from Table 1. The algorithms for construction of the experimental designs were obtained mostly from Das and Giri (1979), Cornell (1981) and Panse and Sukhatme (1978). DESIGNEX is a stand-alone system which requires MS-DOS 3.0 or later version. The programme needs maximum amount of memory available under DOS (out of 640 KB base memory). No memory resident programmes or drivers such as ***Himem. sys*** should be loaded. The package has been accommodated in a single 5.25 inch diskette of 1.2 MB floppy. The same can also be supplied in other formats.

At execution, the system first gathers information about several aspects of the proposed investigation from the experimenter like number of factors, variability in the experimental material, number of treatments, nature of the number of treatments. quantity of experimental material, nature of factors, objective of the experiment, nature of factor levels, number of levels of factors, type of experiment, etc. by means of a number of questions presented in menus. Based on the answers given, the system selects the experimental design and also provides a randomized layout of the design. Some of the terms that shall be encountered while using the DESIGNEX are explained in the Appendix.

With a view to expand the system in later stages, some additional options are included in the present system. Eliminating such options will simplify the present version of the system, but may pose difficulties later. There is a lack of facility for printing the results through the expert system shell. This could not be achieved as the expert system shell GENEX, had no such capability. This facility will be included in the later version of GENEX. The key diskette for running the software was made indispensable, as part of the policy regarding the software protection by ER & DC.

Developing a system of this kind is a premier work, which has high practical uses such as choosing an appropriate experimental design according to the requirements of the investigator and getting the layout plan for setting out the experiments in the field.

5.2. Structure of Knowledge Base

The Knowledge Base of DESIGNEX consists of functions and their descriptions. A function is composed of rules/action, relations, variables, input and output parameters. For creating different experimental designs, several functions are needed. For example, for developing the Completely Randomized Design(CRD), three functions viz., completely-random-design, create-random design and a create-CRD-layout are used. Rules of these three functions are given below.

Complete1y-random-design

(display - text (Completely Randomized Design))
(label : retry)
(if no-of-treatments < 2 then elicit - again no-of- treatments, branch retry)
(if nof-replications is yes then replication = no-of-rep else(rep1ication = (round to int ((12/no-of-treatment) +1))))
(create - random - design no-of-treatments replication)

Create-random-design

(make-empty-list design-list)
(make-empty-list treatment-list)
(make-empty-list layout-design)
(n-o-t = no-of-treatments)
(n-o-r = no-of-replications)
(label : loopl)
(if no-of-treatments == 0 then branch replica)
(put-in-list no-of-treatments in treatment-list)
(decrement no-of-treatments)
(branch loopl)
(label : replica)
(if number-of-replications == 0 then create-CRD-layout n-o-r n-o-t design-list. plot design- list), (plot layout-1ist), skip-the-rest)
(put-in-list treatments-list in design-list)
(decrement number-of-rep1ications)
(branch replica)

Create-CRD-layout

(make-empty-list layout-list)
(treatment = n-o-t)
(temp = flattenfn d-list)

```
(label : loop2)
(make-empty-list layout-design)
(if n-o-r == 0 then skip-the-rest)
(seed = length temp)
(label : loopl)
(if n-o-t == 0 then n-o-t = treatment, put-in-list layout-design in layout-list, decrement n-
o-r, branch loop2)
(pos = remainder (randomfn seed)seed)
(layout = nth pos temp)
(put-in-list layout in layout-design)
(decrement n-o-t)
(temp = delete layout temp)
(decrement seed)
(branch loopl)
```

These three functions are user defined functions. The rules of these function\ also have Genex functions. With these three functions, user gets the completely randomized design format and its layout. In the same manner, several functions are written to develop different other experimental designs and their layout.

Table 1. Experimental designs covered by the expert system DESIGNEX

SERIAL NO.	EXPERIMENTAL DESIGN
1	Completely Randomized Design (CRD)
2	Randomized Complete Block Design (RCBD)
3	Simple Lattice Design
4	Triple Lattice Design
5	Balanced Lattice Design
6	Augmented RCBD
7	CRD - factorial (two factors)
8	CRD - factorial (three factors)
9	CRD - factorial (more than three factors)
10	RCBD - factorial (two factors)
11	RCBD - factorial (three factors)
12	RCBD - partially confounded factorial design
13	Confounded Factorial Design - Balanced Factorial Experiment
14	Split-Plot Design (two factors)
15	Strip-Plot Design
16	Compact Family Block Design
17	Simplex Centroid Design under CRD
18	Simplex Centroid Design under RCBD
19	Simplex Centroid Design compounded with a factorial CRD
20	Simplex Centroid Design compounded with a factorial RCBD
21	Direct Assay - CRD
22	Direct Assay - RCBD
23	Parallel Line Assay - CRD (two doses)
24	Parallel Line Assay - CRD (three doses)
25	Parallel Line Assay - RCBD (two doses)
26	Parallel Line Assay - RCBD (three doses)
27	Parallel Line Assay - Modified BIBD (more than or equal to four doses)
28	Slope Ratio Assay - CRD (two doses)
29	Slope Ratio Assay - CRD (three doses)
30	Slope Ratio Assay - RCBD (two doses)
31	Slope Ratio Assay - RCBD (three doses)
32	Slope Ratio Assay - Modified BIBD (more than or equal to four doses)

5.3 Illustration on the use of DESIGNEX

Illustration on running the expert system DESIGNEX for the case of an experiment to evaluate the efficacy of three doses of bacterial pesticide - *Bacillus thuringiensis*- on two Lepidopteran. pests of teak, namely *Hyblaen puern* and. *Eutectona machaeralis* w th five replications is given below. The experimental material in this case. would be larvae released on one year old teak sa lings. If the larvae are of different stages of development, then the experimental material has to be taken as heterogeneous and blockin can be done based on this factor. The command to invoke the programme is DESIGNEX at the DOS prompt followed by <CR> key. The options are chosen by using the arrow keys (↓ or ↑) and confirmed by pressing the <CR> key. The notation <CR> or "Enter" is used to denote "Press Enter" through out the rest of the manual.

> DESIGNEX

Press <CR> key

DESIGNEX Expert System for designing Experiments in Forestry **Version 1.0** Division of Statistics Kerala Forest Research Institute, Peechi, Thrissur, Kerala. In collaboration with **: CIRA, ER&DC, THIRUVANANTHAPURAM**
DESIGNEX is developed using GENEX General Purpose Expert System Shell of **CIRA, ER &DC,** Thiruvananthapuram - 695 033, KERALA.

Press <CR> key

GENEX - General Purpose Expert System Shell		
run	DESIGNEX	
An Expert System for Designing Experiments in Forestry		
F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 Home End Enter Esc		

GENEX - General Purpose Expert System Shell

'un | | DESIGNEX | DESCRIPTION

This software package helps in choosing a design for scientific investigations and also in getting layout plans of the experimental designs.

F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 HomeEnd Enter Esc

Press <CR> key

GENEX - General Purpose Expert System Shell

run | DESIGNEX | | interactive mode | run mode |

What is the purpose of consulting?

to plan a study
to analyze data

F1 - DIALOGUE mode

F1 F2 F3 F4 F5 F6 F7 F8 F9 HomeEnd Enter Esc

Choose the required option and press <CR> key

GENEX - General Purpose Expert System Shell					
run		DESIGNEX	interactive mode		run mode

What is the type of investigation?

experiment
survey

F1 - DIALOGUE mode

F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 Home End Enter Esc

Choose the required option and press <CR> key

GENEX - General Purpose Expert System Shell					
run		DESIGNEX	interactive mode		run mode

Advice required on

choosing a design
others

F1 - DIALOGUE mode

F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 Home End Enter Esc

Choose the required option and press <CR> key

GENEX - General Purpose Expert System Shell					
run		DESIGNEX	interactive mode		run mode

What is the number of factors?

single

multiple

F1 - DIALOGUE mode

F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 Home End Enter Esc

Choose the required option and press <CR> key

GENEX - General Purpose Expert System Shell					
run		DESIGNEX	interactive mode		run mode

What is the nature of factors?

quantitative

qualitative

quantitative & qualitative

F1 - DIALOGUE mode

F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 Home End Enter Esc

Choose the required option and press <CR> key

GENEX - General Purpose Expert System Shell

run | | DESIGNEX | interactivemode | | runmode |

What are the number of factors?

two
three
more than three

F1- DIALOGUE mode

F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 Home End Enter Esc

Choose the required option and press <CR> key

GENEX - General Purpose Expert System Shell

run | | DESIGNEX | interactive mode | | runmode |

Variability in the experimental material

homoeeneous
heterogeneous

F1- DIALOGUE mode

F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 HomeEnd Enter Esc

Choose the required option and press <CR> key

GENEX - General Purpose Expert System Shell					
run		DESIGNEX	interactive mode		run mode

How many factor levels require large plot size?

for one factor
for two factors
not required

F1 - DIALOGUE mode

F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 Home End Enter Esc

Choose the required option and press <CR> key

GENEX - General Purpose Expert System Shell					
run		DESIGNEX	interactive mode		run mode

Randomized Complete Block Design - Factorial with 2 factors

F1 - DIALOGUE mode

F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 Home End Enter Esc

Press <CR> key

GENEX - General Purpose Expert System Shell

run | | DESIGNEX | interactive mode | | runmode |

What is the number of levels of first factor? within range 1 to 5

2

F1 - DIALOGUE mode

F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 HomeEnd Enter Esc

Enter the number and press <CR> key

GENEX - General Purpose Expert System Shell

run | | DESIGNEX | interactive mode | | runmode |

What is the number of levels of second factor? within range 1 to 5

3

F1 - DIALOGUE mode

F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 HomeEnd Enter Esc

Enter the number and press key

GENEX - General Purpose Expert System Shell					
run		DESIGNEX	interactive mode		run mode

What is the number of replications? within range 2 to 5

5

F1 - DIALOGUE mode

F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 Home End Enter Esc

Enter the number and press <CR> key

GENEX - General Purpose Expert System Shell					
run		DESIGNEX	interactive mode		run mode

Treatment structure

1 - A0 B0
2 - A0 B1
3 - A0 B2
4 - A1 B0
5 - A1 B1
6 - A1 B2

Where A is the first factor
B is the second factor

Randomized Complete Block Design - Factorial with 2 factors.

Press <CR> key

GENEX - General Purpose Expert System Shell

run | | DESIGNEX | interactive mode | | runmode |

rep1 6 2 1 4 5 3

rep4 5 4 2 1 6 3

rep5 5 6 2 1 3 4

rep3 2 1 3 6 4 5

rep2 3 4 6 2 5 1

Randomized Complete Block Design - Factorial with 2 factors - Randomized layout

Press <CR> key

GENEX - General Purpose Expert System Shell

run | | DESIGNEX | interactivemode | | runmode |

Continue problem solving?

no

yes

F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 Home End Enter Esc

Choose the required option and press <CR> key

6. CONCLUSIONS

An expert system by name DESIGNEX was developed using the shell, GENEX. The system was found useful for designing forestry experiments and also for generating field layout plans.

7. REFERENCES

Anonymous 1993. GENEX-General Purpose Expert System Shell. Version 4.0 on MS-DOS. Reference Manual. CIRA, ER&DC, Thiruvananthapuram, India. 300 p.

Cornell, J.A. 1981. Experiments with Mixtures: Designs, Models and the Analysis of Mixture Data. John Wiley and Sons, New York, 305 p.

Das, M.N. and Giri, N.C. 1986. Design and Analysis of Experiments. 2nd ed., Wiley Eastern Ltd., 488 p.

Jayaraman, K. and Rugmini, P. 1990. Statistical techniques in forestry research and forestry. KFRI Research Report 65. Kerala Forest Research Institute, Peechi. 46 p.

Kourtz, P. 1990 Artificial intelligence: a new tool for forest management. Can. J. For. Res., 20,428-437.

Latin, R.X.,Miles, G.E., Rettinger, J.C. and Mitchell, J.R. 1990. An expert system for diagnosing Muskmelon disorders. Plant Disease., 74(1): 83-87.

Panse, V.G. and Sukhatme, P.V. 1978. Statistical Methods for Agricultural Workers. IASRI, New Delhi. 34 p.

Appendix

Some of the terms that shall be encountered while using the DESIGNEX are explained below. The terms are arranged alphabetically.

Augmented design

This design is used in cases where sufficient material may not be available for some treatments (new treatments), but there may be no limitation for other treatments (standard treatments). The design can be obtained by adding different new treatments to each block of any standard randomized complete block design or balanced incomplete block design.

Balanced incomplete block design

An incomplete block design with v treatments distributed over b blocks, each of size k, where k is less than v such that each treatment occurs in r blocks, no treatment occurs more than once in a block and each pair of treatments occurs together in l blocks. is called a balanced incomplete block design.

Bioassay

In a bioassay, the object is to compare the efficacy of two or more substances. or preparations, like drugs, by using the responses produced by them on suitable living organisms. In general, there are two types of assays, direct assays and indirect assays. In direct assays the response is pre-assigned as the death of experimental subjects as a result of application of each preparation. In indirect assays, the relationship between the dose and response of each preparation is ascertained first and the dose corresponding to a given response is obtained from the relation worked out. For linearizing the relationship between the dose and response, the doses are transformed to a different scale. Indirect assays further split into parallel line assays and slope ratio assays. In parallel line assays the doses of each of the preparations are taken in geometric progression but at the same time evenly distributed in the transformed scale in which the dose-response relation is linear. In slope ratio assays, equispaced doses of both the preparations are taken. Both direct and indirect assay experiments can be laid out in one of the standard designs like completely randomized design, randomized complete block design etc. When the number of doses is large (more than or equal to 4), it may not be possible to get homogeneous groups of experimental units for adopting randomized block designs. In such situations, one goes for incomplete block designs. This is termed as modified BIBD.

Block

Block refers to a group of experimental units under different treatments, the experimental unit in a block being so far as possible homogeneous.

Compact family block design

This is a design used in plant breeding trials wherein the progenies can be grouped into distinct families. Each block is first divided into main plots and families are allotted randomly to the main plots. The progenies within each family are then allotted randomly to the subplots within each of the main plots.

Completely randomized design

It is an experimental design in which the treatments are allocated to the experimental units purely on a chance basis without any grouping of the experimental units.

Confounded factorial designs

When the number of factors, or the number of levels of the factors in a factorial experiment is large, the number of treatment combinations in the experiment will be huge, giving rise to problems of heterogeneity within a block. In such situations one goes for a confounded factorial design. Confounding is a design technique for arranging a complete factorial experiment in blocks, where the block size is smaller than the number of treatment combinations. The technique causes loss of information on some factorial effects (usually high-order interactions) but unconfounded effects are estimated with higher precision than in a complete block design.

Experiment

Experiments serve to test hypotheses under controlled conditions.

Experimental material

Objects with which an experiment is conducted is termed as experimental material.

Factor

A factor denotes a variable under examination in an experiment as a possible cause of variation.

Factorial experiment

An experiment designed to examine the effects of multiple factors, wherein the levels of more than one factor are allowed to vary simultaneously is known as factorial experiment. The treatments are formed by taking all possible combinations of the levels of the different factors included in the experiment. The factorial experiment can itself be laid out in one of the standard designs like completely randomized design or randomized completely block design.

Heterogeneity

The property of one or more samples or population which implies that they are not identical in respect of some or all of their attributes.

Homogeneity

The property of one or more samples or population which implies that they are identical in respect of some or all of their attributes.

Interaction

Interaction is a measure of the extent to which the effect of one factor changes as the level(s) of another or other factor changes.

Lattice design

It is an experimental design used in cases where the number of treatments is a perfect square, say k^2. A lattice design with 2 replications is known as simple lattice design. one with 3 replications is known as triple lattice design and one with $k+1$ replications is known as balanced lattice design. Depending upon the availability of the experimental material, any one of lattice designs can be chosen. In DESIGNEX, the layout of the lattice designs is available for number of treatments equal to 9, 16, 25, 49, 64 and 81.

Main effect

Main effect is an estimate of the effect of an experimental variable or treatment measured independently of other treatments which may form part of the experiment.

Plot

In experimental designs, plot refers to the basic unit of the experimental material, e.g.. a single tree or part of a tree, a sample of seeds, etc.

Precision

Precision is the closeness with which an estimate approaches the average of a large series of estimates made under similar conditions.

Qualitative

The individuals comprising the material under consideration are distinguished by some quality or attribute, e.g., sex, nationality, etc. If the levels of a quantitative factor is prefixed by the experimenter, then the factor is to be taken as qualitative.

Quantitative

The individuals comprising the material under consideration are distinguished by a quantitative factor, such as measurement or count, e.g., height of a plant, yield of a crop, etc.

Randomized complete block design

An experimental design in which each block contains a complete replicate of the treatments which are allocated to the various plots within the blocks in a random manner, is termed as randomized complete block design.

Replication

Applying a treatment or a set of treatments more than once, so as to increase the accuracy of estimates of the treatment effects and to provide an estimate of the error variance is the concept of replication.

Response surface

A response surface is the geometrical configuration of a factor response relationship. It depicts how the response variable is related to the levels of input variab1es.A factor-response relationship is linear if the change in response is constant throughout the whole range of levels of input variable under consideration. To fit a linear relation, the minimum number of levels of each of the factors is 2. Factor-response relationship is quadratic if the relationship can be expressed as a second degree polynomial. To fit a quadratic relation, the minimum number of levels of each of the factors is 3.

Simplex centroid design

It is a design for experiments with mixtures, wherein the levels of factors are proportions taking values in the range of 0 to 1. The levels of different factors in a treatment combination would add up to 1 in such cases. Because of this restriction, the parameters of the corresponding response function (say polynomial) cannot be estimated uniquely. The polynomial response function is then redefined with changes in the parameter expression which is then called a canonical polynomial. In a q component simplex centroid design, the number of design points is 2^q-1. The design consists of every nonempty subset of the q components, but only on mixtures in which the components present appear in equal proportions. These points in an experiment can be laid out in completely randomized design or randomized complete block design.

Simplex centroid design compounded with a factorial

This design is used when levels of certain factors in a factorial experiment are proportions. For instance, an experimental arrangement of a q component simplex centroid design, compounded with a factorial design having n factors, each at 2 levels is def'ined as 2^n factorial setup at each of the 2^q-1 points of the simplex centroid, or, as a group of 2^q-1 mixtures set up at each combination of the factor levels of a 2^n factorial. In an actual experiment the number of design points is 2^n (2^q-1). These design points can be laid out in completely randomized design or randomized complete block design in the experiment.

Split plot design

In a two factor experiment, if levels of one of the factors require large plot size, one can resort to a split plot arrangement, wherein one whole replication is divided into a set of main plots and each main plot is divided into subplots. The factor for which the levels reyuire larger plot size is put in the main plots and the other in the subplots. Thus, each main plot becomes a block for the subplot treatments. In a split plot design, the subplot factor and its interaction with the main plot factor are estimated with more precision than that of the main plot factor.

Strip plot design

In a two factor experiment, when both the factors require large plot size, one can resort to strip plot design. In a strip plot design, the experimental units are grouped in two directions (horizontal and vertical) by the two factors within a replication. The factor which is applied to horizontal plots is called horizontal factor and the other which is applied to vertical plots is called vertical factor, The horizontal and vertical plots are always perpendicular to each other. However, there is no relationship between their sizes. unlike the case of main plot and subplot of the split plot design. The intersection plot is, of course, the smallest. Thus in a strip plot design, the degrees of precision associated with the main effects of both factors are sacrificed in order to improve the precision of the interaction effect.

Survey

Sample survey is an examination of samples of observations from a population which exists in its own way such that the sample can adequately represent and accurately interpret the population.

Treatment

The different procedures under comparison in an experiment denote the different treatments. E.g., in an agricultural experiment, the different varieties of a crop, the different manures etc. form treatments.