PART TEN

# Evaluating Performance

# 30

# The Problem of Evaluation

Early in the development of MYCIN we felt the need to assess formally the program's performance. By 1973 we had already run perhaps a hundred cases of bacteremia through the system, revising the knowledge base as needed whenever problems were discovered. At the weekly project meetings Cohen and Axline were increasingly impressed by the validity of the program's recommendations, and they encouraged the design of an experiment to assess its performance on randomly selected cases of bacteremic patients. There was a uniform concern that it would be inadequate to assess (or report) the work on the basis of anecdotal accolades alone—an informal approach to evaluation for which many efforts in both AI and medical computer science had been criticized.

## 30.1 Three Evaluations of MYCIN

Shortliffe accordingly designed and executed an experiment that was reported as a chapter in his dissertation (Shortliffe, 1974). Five faculty and fellows in the Stanford Division of Infectious Diseases were asked to review and critique 15 cases for which MYCIN had offered therapy advice. Each evaluator ran the first of the 15 cases through MYCIN himself (in order to get a feeling for how the program operated) and was then given printouts showing the questions asked and the advice generated for each of the other 14 cases. Questions were inserted at several places in the typescripts so that we could assess a variety of features of the program:

- its ability to decide whether a patient required treatment;
- its ability to determine the significance of isolated organisms;
- its ability to determine the identity of organisms judged significant;
- its ability to select therapy to cover for the list of most likely organisms;
- its overall consultative performance.

The design inherently assumed that the opinions of recognized experts provided the "gold standard" against which the program's performance should be assessed. For reasons outlined below, other criteria (such as the actual organisms isolated or the patient's response to therapy) did not seem appropriate. Despite the encouraging results of this experiment (hereafter referred to as Study 1), several problems were discovered during its execution:

- The evaluators complained that they could not get an adequate "feel" for the patients by merely reading a typescript of the questions MYCIN asked (and they therefore wondered how the program could do so).
- Because the evaluators knew they were assessing a computer program, there was evidence that they were using different (and perhaps more stringent) criteria for assessing its performance than they would use in assessing the recommendations of a human consultant.
- MYCIN's "approval rating" of just under 75% was encouraging but intuitively seemed to be too low for a truly expert program; yet we had no idea how high a rating was realistically achievable using the gold standard of approval by experts;
- The time required from evaluators was seen to be a major concern; the faculty and fellows agreed to help with the study largely out of curiosity, but they were all busy with other activities and some of them balked at the time required to thoroughly consider the typescripts and treatment plans for all 15 cases.
- Questions were raised regarding the validity of a study in which the evaluators were drawn from the same environment in which the program was developed; because of regional differences in prescribing habits and antimicrobial sensitivity patterns, some critics urged a study design in which MYCIN's performance in settings other than Stanford could be assessed.

Many of these problems were addressed in the design of our second study, also dealing with bacteremia, which was undertaken in the mid-1970s and for which a published report appeared in 1979 (Yu et al., 1979a). This time the evaluators were selected from centers around the country (five from Stanford, five from other centers) and were paid a small honorarium in an effort to encourage them to take the time required to fill out the evaluation forms. Because the evaluators did not have an opportunity to run the MYCIN program themselves, we deemphasized the actual appearance of a MYCIN typescript in this study (hereafter referred to as Study 2). Instead, evaluators were provided with copies of each of the 15 patients' charts up to the time of the therapy decisions (with suitable precautions taken to preserve patient anonymity). They once again knew they were evaluating a computer program, however. In addition, although the

forms were designed to allow evaluators to fill them out largely by using checklists, the time required to complete them was still lengthy if the physician was careful in the work, and there were once again long delays in getting the evaluation forms back for analysis. In fact, despite the "motivating honorarium," some of the evaluators took more than 12 months to return the booklets.

Although the MYCIN knowledge base for bacteremia had been considerably refined since Study 1, we were discouraged to find that the results of Study 2 once again showed about 75% overall approval of the program's advice. It was clear that we needed to devise a study design that would "blind" the evaluators to knowledge of which advice was generated by MYCIN and that would simultaneously allow us to determine the overall approval ratings that could be achieved by experts in the field. We began to wonder if the 75% figure might not be an upper limit in light of the controversy and stylistic differences among experts.

As a result, our meningitis study (hereafter referred to as Study 3) used a greatly streamlined design to encourage rapid turnaround in evaluation forms while keeping evaluators unaware of what advice was proposed by MYCIN (as opposed to other prescribers from Stanford). Study 3 is the subject of Chapter 31, and the reader will note that it reflects many of the lessons from the first two studies cited above. With the improved design we were able to demonstrate formally that MYCIN's advice was comparable to that of infectious disease experts and that 75% is in fact *better* than the degree of agreement that could generally be achieved by Stanford faculty being assessed under the same criteria.

In the next section we summarize some guidelines derived from our experience. We believe they are appropriate when designing experiments for the evaluation of expert systems. Then, in the final section of this chapter, we look at some previously unpublished analyses of the Study 3 data. These demonstrate additional lessons that can be drawn and on which future evaluative experiments may build.

## 30.2 A Summary of Evaluation Considerations

The three MYCIN studies, plus the designs for ONCOCIN evaluations that are nearing completion, have taught us many lessons about the validation of these kinds of programs. We summarize some of those points here in an effort to provide guidelines of use to others doing this kind of work.[1]

[1]Much of this discussion is based on Shortliffe's contribution to Chapter 8 of *Building Expert Systems*, edited by R. Hayes-Roth, D. Lenat, and D. Waterman (Hayes-Roth, Waterman and Lenat, 1983).

### 30.2.1 Dependence on Task, System, Goals, and Stage of Development

Most computing systems are developed in response to some human need, and it might therefore be logical to emphasize the system's response to that need in assessing whether it is successful. Thus there are those who would argue that the primary focus of a system evaluation should be on the task for which it was designed and the quality of its corresponding performance. Other aspects warranting formal evaluation are often ignored. It must accordingly be emphasized that there are diverse components to the evaluation process. We believe that validation is most appropriately seen as occurring in stages as an expert system develops over time.

The MYCIN work, however, has forced us to focus our thinking on the evaluation of systems that are ultimately designed to perform a real-world task, typically to be used by persons who are not computer scientists. Certainly one of our major goals has been the development of a useful system that can have an impact on society by becoming a regularly used tool in the community for which it is designed. Although we have shown in earlier chapters that many basic science problems typically arise during the development of such systems, in this section we will emphasize the staged assessment of the developing tool (rather than techniques for measuring its scientific impact as a stimulus to further research). We have organized our discussion by looking at the "what?", "when?", and "how?" of evaluating expert systems.

### 30.2.2 What to Evaluate?

As mentioned above, at any stage in the development of a computing system several aspects of its performance could be evaluated. Some are more appropriate than others at a particular stage. However, by the time a system has reached completion it is likely that every aspect will have warranted formal assessment.

#### Decisions/Advice/Performance

Since accurate, reliable advice is an essential component of an expert consultation system, it is usually the area of greatest research interest and is logically an area to emphasize in evaluation. However, the mechanisms for deciding whether a system's advice is appropriate or adequate may be difficult to define or defend, especially since expert systems tend to be built precisely for those domains in which decisions are highly judgmental. It is clear that no expert system will be accepted by its intended users if they fail to be convinced that the decisions made and the advice given are pertinent and reliable.

### Correct Reasoning

Not all designers of expert systems are concerned about whether their program reaches decisions in a "correct" way, so long as the advice that it offers is appropriate. As we have indicated, for example, MYCIN was not intended to simulate human problem solving in any formal way. However, there is an increasing realization that expert-level performance may require heightened attention to the mechanisms by which human experts actually solve the problems for which the expert systems are being built. It is with regard to this issue that the interface between knowledge engineering and psychology is the greatest, and, depending on the motivation of the system designers and the eventual users of the expert program, some attention to the mechanisms of reasoning that the program uses may be appropriate during the evaluation process. The issue of deciding whether or not the reasoning used by the program is "correct" will be discussed further below.

### Discourse (I/O Content)

Knowledge engineers now routinely accept that parameters other than correctness will play major roles in determining whether or not their systems are accepted by the intended users (see Chapter 32). The nature of the discourse between the expert system and the user is particularly important. Here we mean such diverse issues as:

- the choice of words used in the questions and responses generated by the program;
- the ability of the expert system to explain the basis for its decisions and to customize those explanations appropriately for the level of expertise of the user;
- the ability of the system to assist the user when he or she is confused or wants help; and
- the ability of the expert system to give advice and to educate the user in a congenial fashion so that the frequently cited psychological barriers to computer use are avoided.

It is likely that issues such as these are as important to the ultimate success of an expert system as is the quality of its advice. For this reason such issues also warrant formal evaluation.

### Hardware Environment (I/O Medium)

Although some users, particularly when pressed to do so, can become comfortable with a conventional typewriter keyboard to interact with computers, this is a new skill for other potential users and frequently not one

they are motivated to learn. For that reason we have seen the development of light pen interfaces, touch screens, and specialized keypads, any of which may be adequate to facilitate simple interactions between users and systems. Details of the hardware interface often influence the design of the system software as well. The intricacies of this interaction cannot be ignored in system evaluation, nor can the mundane details of the user's reaction to the terminal interface. Once again, it can be difficult to design evaluations in which dissatisfaction with the terminal interface is isolated as a variable, independent of discourse adequacy or decision-making performance. As we point out below, one purpose of staged evaluations is to eliminate some variables from consideration during the evolution of the system.

#### Efficiency

Technical analyses of system behavior are generally warranted. Underutilized CPU power or poorly designed methods for accessing disk space, for example, may introduce resource inefficiencies that severely limit the system's response time or cost effectiveness. Inefficiencies in small systems are often tolerable to users, but will severely limit the potential for those systems to grow and still remain acceptable.

#### Cost Effectiveness

Finally, and particularly if it is intended that an expert system become a widely used product, some detailed evaluation of its cost effectiveness is necessary. A system that requires an excessive time commitment by the user, for example, may fail to be accepted even if it excels at all the other tasks we have mentioned. Few AI systems have reached this stage in system evolution, but there is a wealth of relevant experience in other computer science areas. Expert systems must be prepared to embark on similar studies once they reach an appropriate stage of development.

### 30.2.3 When to Evaluate?

The evaluation process is a continual one that should begin at the time of system design, extend in an informal fashion through the early stages of development, and become increasingly formal as a developing system moves toward real-world implementation. It is useful to cite nine stages of system development, which summarize the evolution of an expert system.[2] They are itemized in Table 30-1 and discussed in some detail below.

[2]These implementation steps are based on a discussion of expert systems in Shortliffe and Davis (1975).

**TABLE 30-1 Steps in the Implementation of an Expert System**

1. Top-level design with definition of long-range goals
2. First version prototype, showing feasibility
3. System refinement in which informal test cases are run to generate feedback from the expert and from users
4. Structured evaluation of performance
5. Structured evaluation of acceptability to users
6. Service functioning for extended period in prototype environment
7. Follow-up studies to demonstrate the system's large-scale usefulness
8. Program changes to allow wide distribution of the system
9. General release and distribution with firm plans for maintenance and updating

As mentioned above, it is important for system designers to be explicit about their long-range goals and motives for building an expert system. Thus the first stage of a system's development (Step 1), the initial design, should be accompanied by explicit statements of what the measures of the program's success will be and how failure or success will be evaluated. It is not uncommon for system designers to ignore this issue at the outset. If the evaluation stages and long-range goals are explicitly stated, however, they will necessarily influence the early design of the expert system. For example, if explanation capabilities are deemed to be crucial for the user community in question, this will have important implications for the system's underlying knowledge representation.

The next stage (Step 2) is a demonstration that the design is feasible. At this stage there is no attempt to demonstrate expert-level performance. The goal is, rather, to show that there is a representation scheme appropriate for the task domain and that knowledge-engineering techniques can lead to a prototype system that shows some reasonable (if not expert) performance on some subtask of that domain. An evaluation of this stage can be very informal and may simply consist of showing that a few special cases can be handled by the prototype system. Successful handling of the test cases suggests that with increased knowledge and refinement of the reasoning structures a high-performance expert system is possible.

The third stage (Step 3) is as far as many systems ever get. This is the period in which informal test cases are run through the developing system, the system's performance is observed, and feedback is sought from expert collaborators and potential end users. This feedback serves to define the major problem areas in the system's development and guides the next iteration in system development. This iterative process may go on for months or years, depending on the complexity of the knowledge domain, the flexibility of the knowledge representation, and the availability of techniques adequate to cope with the domain's specific control or strategic processes. One question is constantly being asked: how did this system do on this case? Detailed analyses of strengths and weaknesses lead back to further research; in this sense evaluation is an intrinsic part of the system development process.

Once the system is performing well on most cases with which it is presented, it is appropriate to turn to a more structured evaluation of its decision-making performance. This evaluation can be performed without assessing the program's actual utility in a potential user's environment. Thus Step 4 is undertaken if the test cases being used in Step 3 are found to be handled with skill and competence, and there accordingly develops a belief that a formal randomized study will show that the system is capable of handling almost any problem from its domain of expertise. Only a few expert systems have reached this stage of evaluation. The principal examples are studies of the PROSPECTOR program developed at SRI International (Gaschnig, 1979) and the MYCIN studies described earlier in this chapter. It should be emphasized that a formal evaluation with randomized case selection may show that the expert system is in fact not performing at an expert level. In this case, new research problems or knowledge requirements are defined, and the system development returns to Step 3 for additional refinement. A successful evaluation at Step 4 is desirable before a program is introduced into a user environment.

The fifth stage (Step 5), then, is system evaluation in the setting where the intended users have access to it. The principal question at this stage is whether or not the program is acceptable to the users for whom it was intended. Essentially no expert systems have been formally assessed at this stage. The emphasis in Step 5 is on the discourse abilities of the program, plus the hardware environment that is provided. If expert-level performance has been demonstrated at Step 4, failure of the program to be accepted at Step 5 can be assumed to be due to one of these other human factors.

If a system is formally shown to make good decisions and to be acceptable to users, it is appropriate to introduce it for extended periods in some prototype environment (Step 6). This stage, called *field testing*, is intended largely to gain experience with a large number of test cases and with all the intricacies of on-site performance. Careful attention during this stage must be directed toward problems of scale: i.e., what new difficulties will arise when the system is made available to large numbers of users outside of the direct control of the system developers? Careful observation of the program's performance and the changing attitudes of those who interact with it are important at this stage.

After field testing, it is appropriate to begin follow-up studies to demonstrate a system's large-scale usefulness (Step 7). These formal evaluations often require measuring pertinent parameters before and after introducing the system into a large user community (different from the original prototype environment). Pertinent issues are the system's efficiency, its cost effectiveness, its acceptability to users who were not involved in its early experimental development, and its impact on the execution of the task with which it was designed to assist. During Step 7 new problems may be discovered that require attention before the system can be distributed (Step

8). These may involve programming changes or modifications required to allow the system to run on a smaller or exportable machine.

Finally, the last stage in system development is general release as a marketable product or in-house tool (Step 9). Inherent at this stage are firm plans for maintaining the knowledge base and keeping it current. One might argue that the ultimate evaluation takes place at this stage when it is determined whether or not the system can succeed in broad use. However, a system's credibility is likely to be greater if good studies have been done in the first eight stages so that there are solid data supporting any claims about the quality of the program's performance.

### 30.2.4 How to Evaluate?

It would be folly to claim that we can begin to suggest detailed study designs for all expert systems in a single limited discussion. There is a wealth of information in the statistical literature, for example, regarding the design of randomized controlled trials, and much of that experience is relevant to the design of expert system evaluations. Our intention here, therefore, is to concentrate on those issues that complicate the evaluation of expert systems in particular and to suggest pitfalls that must be considered during study design.

We also wish to distinguish between two senses of the term *evaluation.* In computer science, system evaluation often is meant to imply optimization in the technical sense—timing studies, for example. Our emphasis, on the other hand, is on a system's performance at the specific consultation task for which it has been designed. Unlike many conventional programs, expert systems do not deal with deterministic problems for which there is clearly a right or wrong answer. As a result, it is often not possible to demonstrate in a straightforward fashion that a system is "correct" and then to concentrate one's effort on demonstrating that it reaches the solution to a problem in some optimal way.

#### Need for an Objective Standard

Evaluations require some kind of "gold standard"—a generally accepted correct answer with which the results of a new methodology can be compared. In the assessment of new diagnostic techniques in medicine, for example, the gold standard is often the result of an invasive procedure that physicians hope to be able to avoid, even though it may be 100% accurate (e.g., operative or autopsy results, or the findings on an angiogram). The sensitivity and specificity of a new diagnostic liver test based on a blood sample, for example, can best be assessed by comparing test results with the results of liver biopsies from several patients who also had the blood test; if the blood test is thereby shown to be a good predictor of

the results of the liver biopsy, it may be possible to avoid the more invasive procedure in future patients. The parallel in expert system evaluation is obvious; if we can demonstrate that the expert system's advice is comparable to the gold standard for the domain in question, it may no longer be necessary to turn to the gold standard itself if it is less convenient, less available, or more expensive.

### Can the Task Domain Provide a Standard?

In general there are two views of how to define a gold standard for an expert system's domain: (1) what eventually turns out to be the "correct" answer for a problem, and (2) what a human expert says is the correct answer when presented with the same information as is available to the program. It is unfortunate that for many kinds of problems with which expert systems are designed to assist, the first of these questions cannot be answered or is irrelevant. Consider, for example, the performance of MYCIN. One might suggest that the gold standard in its domain should be the identity of the bacteria that are ultimately isolated from the patient, or the patient's outcome if he or she is treated in accordance with (or in opposition to) the program's recommendation. Suppose, then, that MYCIN suggests therapy that covers for four possibly pathogenic bacteria but that the organism that is eventually isolated is instead a fifth rare bacterium that was totally unexpected, even by the experts involved in the case. In what sense should MYCIN be considered "wrong" in such an instance? Similarly, the outcome for patients treated for serious infections is not 100% correlated with the correctness of therapy; patients treated in accordance with the best available medical practice may still die from fulminant infection, and occasionally patients will improve despite inappropriate antibiotic treatment. Accordingly, we said that MYCIN performed at an expert level and was "correct" if it agreed with the experts, even if both MYCIN and the experts turned out to be wrong. The CADUCEUS program has been evaluated by comparing the diagnoses against those published on selected hard cases from the medical literature (Miller et al., 1982).

### Are Human Experts Evaluated?

When domain experts are used as the objective standard for performance evaluation, it is useful to ask whether the decisions of the experts themselves are subjected to rigorous evaluations. If so, such assessments of human expertise may provide useful benchmarks against which to measure the expertise of a developing consultation system. An advantage of this approach is that the technique for evaluating experts is usually a well-accepted basis for assessing expertise and thus lends credibility to an evaluation of the computer-based approach.

### Informal Standards

Typically, however, human expertise is accepted and acknowledged using less formal criteria, such as level of training, recommendations of previous clients, years of experience in a field, number of publications, and the like. [Recently, Johnson et al. (1981) and Lesgold (1983) have studied measures of human expertise that are more objective.] Testimonials regarding the performance of a computer program have also frequently been used as a catalyst to the system's dissemination, but it is precisely this kind of anecdotal selling of a system against which we are arguing here. Many fields (e.g., medicine) will not accept technological innovation without rigorous demonstration of the breadth and depth of the new product's capabilities. Both we and the PROSPECTOR researchers encountered this cautious attitude in potential users and designed their evaluations largely in response to a perceived need for rigorous demonstrations of performance.

### Biasing and Blinding

In designing any evaluative study, considerations of sources of bias are of course important. We learned this lesson when evaluating MYCIN, and, as mentioned earlier, this explains many of the differences between the bacteremia evaluation (Study 2) and the meningitis study (Study 3). Many comments and criticisms from Study 2 evaluators reflected biases regarding the proper role for computers in medical settings (e.g., "I don't think the computer has an adequate sense of how sick this patient is. You'd have to see a patient like this in order to judge."). As a result, Study 3 mixed MYCIN's recommendations with a set of recommendations from nine other individuals asked to assess the case (ranging from infectious disease faculty members to a medical student). When national experts later gave opinions on the appropriateness of therapeutic recommendations, they did not know which proposed therapy (if any) was MYCIN's and which came from the faculty members. This "blinded" study design removed an important source of potential bias, and also provided a sense of where MYCIN's performance lay along a range of expertise from faculty to student.

### Controlling Variables

As we pointed out in the discussion of *when* to evaluate an expert system, one advantage of a sequential set of studies is that each can assume the results of the experiments that preceded it. Thus, for example, if a system has been shown to reach optimal decisions in its domain of expertise, one can assume that the system's failure to be accepted by its intended users in an experimental setting is a reflection of inadequacies in an aspect of the system *other* than its decision-making performance. One key variable that could account for system failure can be "removed" in this way.

### Realistic Standards of Performance

Before assessing the capabilities of an expert system, it is necessary to define the minimal standards that are acceptable for the system to be called a success. It is ironic that in many domains it is difficult to decide what level of performance qualifies as expert. Thus it is important to measure the performance of human experts in a field if they are assessed by the same standards to be used in the evaluation of the expert system. As we noted earlier, this point was demonstrated in the MYCIN evaluations. In Studies 1 and 2, MYCIN's performance was approved by a majority of experts in approximately 75% of cases, a figure that seemed disappointingly low to us. We felt that the system should be approved by a majority in at least 90% of cases before it was made available for actual clinical use. The blinded study design for the subsequent meningitis evaluation (Study 3), however, showed that even infectious disease faculty members received at best a 70–80% rating from other experts in the field. Thus the 90% figure originally sought may have been unrealistic in that it inadequately reflected the extent of disagreement that can exist even among experts in a field such as clinical medicine.

### Sensitivity Analysis

A special kind of evaluative procedure that is pertinent for work with expert systems is the analysis of a program's sensitivity to slight changes in knowledge representation, inference weighting, etc. Similarly, it may be pertinent to ask which interactive capabilities were necessary for the acceptance of an expert consultant. One approach to assessing these issues is to compare two versions of the system that vary the feature under consideration. An example of studies of this kind are the experiments that we did to assess the certainty factor model. As is described in Chapter 10 (Section 10.3), Clancey and Cooper showed that the decisions of MYCIN changed minimally from those reported in the meningitis evaluation (Chapter 31) over a wide range of possible CF intervals for the inferences in the system. This sensitivity analysis helped us decide that the details of the CF's associated with rules mattered less than the semantic and structural content of the rules themselves.

### Interaction of Knowledge: Preserving Good Performance When Correcting the Bad

An important problem, discussed in Chapter 7, can be encountered when an evaluation has revealed system deficiencies and new knowledge has been added to the system in an effort to correct these. In complex expert systems, the interactions of new knowledge with old can be unanticipated and

lead to detrimental effects on problems that were once handled very well by the system. An awareness of this potential problem is crucial as system builders iterate from Step 3 to Step 4 and back to Step 3 (see Table 30-1). One method for protecting against the problem is to keep a library of old cases available on-line for batch testing of the system's decisions. Then, as changes are made to the system in response to the Step 4 evaluations of the program's performance, the old cases can be run through the revised version to verify that no unanticipated knowledge interactions have been introduced (i.e., to show that the program's performance on the old cases does not deteriorate).

### Realistic Time Demands on Evaluators

A mundane issue that must be considered anyway, since it can lead to failure of a study design or, at the very least, to unacceptable delays in completing the program's assessment, is the time required for the evaluators to judge the system's performance. If expert judgments are used as the gold standard for adequate program performance, the opinions of the experts must be gathered for the cases used in the evaluation study. A design that picks the most pertinent two or three issues to be assessed and concentrates on obtaining the expert opinions in as easy a manner as possible will therefore have a much better chance of success. We have previously mentioned the one-year delay in obtaining the evaluation booklets back from the experts who had agreed to participate in the Study 2 bacteremia evaluation. By focusing on fewer variables and designing a checklist that allowed the experts to assess program performance much more rapidly, the meningitis evaluation was completed in less than half that time (Chapter 31).

## 30.3 Further Comments on the Study 3 Data

When the Study 3 data had been analyzed and published (Chapter 31), we realized there were still several lingering questions. The journal editors had required us to shorten the data analysis and discussion in the final report. We also had asked ourselves several questions regarding the methodology and felt that these warranted further study.

Accordingly, in 1979 Reed Letsinger (then a graduate student in our group) undertook an additional analysis of the Study 3 data. What follows is largely drawn from an internal memo that he prepared to report his findings. The reader should be familiar with Chapter 31 before studying the sections below.

### 30.3.1 Consistency of the Evaluators

The eight national evaluators in Study 3 could have demonstrated internal inconsistency in two ways. Since each one was asked first to indicate his own decision, he could be expected to judge as acceptable any of the prescribers' decisions that were identical to his own. The first type of inconsistency would occur if this expectation were violated. Among the 800 judgments in the Study 3 data (8 evaluators × 10 prescribers × 10 patients), 15 instances of this type of inconsistency occurred. Second, since several prescribers would sometimes make the same decision regarding a patient, another form of inconsistency would occur if an evaluator were to mark identical treatments for the same patient differently for different prescribers. Since the evaluators had no basis for distinguishing among the subjects (prescribers), such discrepancies were inherently inconsistent. Twenty-two such instances occurred in the Study 3 data set.

These numbers indicate that 37 out of the 800 data points (4.6%) could be shown to be in need of correction on the basis of these two tests. Such a figure tells us something about the reliability of the data—clearly pertinent in assessing the study results. We have wondered about plausible explanations for these kinds of inconsistencies. One is that the evaluators were shown both the *drugs* recommended by the prescribers *and* the recommended *doses*. They were asked to base their judgment of treatment acceptability on drug selection alone, but we did ask separately for their opinion on dosage to help us assess the adequacy of MYCIN's dosing algorithms (see Chapter 19). It appeared in retrospect, however, that the evaluators sometimes ignored the instructions and discriminated between two therapy prescriptions that differed only in the doses of the recommended drugs. These judgments are thus only inconsistent in the sense that they reflect judgments that the evaluators were not supposed to be making. The problem reflects the inherent tension between our wanting to get as much possible information from evaluators and the risks in introducing new variables or data that may distract evaluators from the primary focus of the study. Another methodologic point here is that such design weaknesses may be uncovered by making some routine tests for consistency.

### 30.3.2 Agreement Among Evaluators

The tendency of the experts to agree with one another has a direct impact on the power of the study to discriminate good performance from bad. Consider two extreme cases. At one end is the case where on the average the evaluators agree with each other just as much as they disagree. This means that on each case the prescribers would tend to get scores around the midpoint—in the case of the MYCIN study, around 4 out of 8. The cumulative scores would then cluster tightly around the midpoint of the

possible range, e.g., around 40 out of 80. The differences between the quality of performance of the various subjects would be "washed out," the scores would all be close to one another, and consequently, it would be very unlikely that any of the differences between scores would be significant. At the other extreme, if the evaluators always agreed with each other, the only "noise" in the data would be contributed by the choice of the sample cases. Intermediate amounts of disagreement would correspondingly have intermediate effects on the variability of the scores, and hence on the power of the test to distinguish the performance capabilities of the subjects.

A rough preliminary indication of the extent of this agreement can be derived from the MYCIN data. A judgment situation consists of a particular prescriber paired with a particular case. Thus there are 100 judgment situations in the present study, and each receives a score between 0 and 8, depending on how many of the evaluators found the performance of the subject acceptable on the case. The range between 0 and 8 is divided into three equal subranges, 0 to 2, 3 to 5, and 6 to 8. A judgment situation receiving a score in the first of these ranges may be said to be generally unacceptable, while those receiving scores in the third range are generally acceptable. The situations scoring in the middle range, however, cannot be decided by a two-thirds majority rule, and so may be considered to be undecided due to the evaluators' inability to agree. It turns out that 53 out of the 100 judgment situations were undecided in this sense in the MYCIN study.

For a more accurate indication of the level of this disagreement, the evaluators can be paired in all possible combinations, and the percentage of judgment situations in which they agree can be calculated. The mean of this percentage across all pairs of evaluators reflects how often we should expect two experts to agree on the question of whether or not the performance of a prescriber is acceptable (when the experts, the prescriber, and the case are chosen from populations for which the set of evaluators, the set of subjects, and the set of cases used in the study are representative samples). In the MYCIN study, this mean was 0.591. Thus, if the evaluators, prescribers, and cases used in this study are representative, we would in general expect that if we choose two infectious disease experts and a judgment situation at random on additional cases, the two experts will disagree on the question of whether or not the recommended therapy is acceptable 4 out of every 10 times!

Before such a number can be interpreted, more must be known about the pattern of agreement. One question is how the disagreement was distributed across the subjects and across the cases. It turns out that the variation across subjects was remarkably low for the MYCIN data, with a standard deviation of less than 6 percentage points. The standard deviation across cases was slightly higher—just under 10 percentage points. Very little of the high level of disagreement among the graders can be attributed to the idiosyncracies of a few subjects or of a few cases. If it had turned

out that a large amount of the disagreement focused on a few cases or a few subjects, they could have been disregarded, and the power of the study design increased.

A second question that can be raised is to what extent the disagreements result from differing tolerance levels among the different evaluators for divergent recommendations. A quick and crude measure of this tolerance level is simply the percentage of favorable responses the evaluators gave. The similarity between the tolerance levels of two graders can be measured by the difference between those percentages. It is then possible to rank all the pairs of evaluators in terms of the degree of similarity of their tolerance levels, just as it is possible to rank pairs of evaluators by their agreements. The extent to which the tendency of the evaluators to agree or disagree with one another can be explained by the variation in their tolerance levels can be measured by the correlation between these two rankings. With the MYCIN study, the Spearman rank correlation coefficient turns out to be 0.0353 with no correction for ties. This is not significantly greater than 0. If there had been a significant correlation, the scores given by the evaluators could have been weighted in order to normalize the effects due to different tolerance levels. The actual disagreement among the evaluators would then have been reduced.

A third possibility is that different groups of experts represent different schools of thought on solving the type of problems represented in our sample. If so, there should be clusters of evaluators, all of whose members agree with each other more than usual, while members of different clusters tend to disagree more than usual. There was some slight clustering of this sort in the MYCIN data. Evaluators 1, 3, and 4 all agreed with each other more often than the mean of 0.591, as did 2 and 6, and matching any member of the first group with any member of the second gives an agreement of less than the mean. However, evaluator 8 agreed with all five of these evaluators more than 0.591. These clusterings are probably real, but they cannot account for very much of the tendency of the evaluators to disagree. If significant clustering had been uncovered, the data could have been reinterpreted to treat the different "schools" of experts as additional variables in the analysis. Within each of these "schools," the agreement would then have been considerably increased.

In retrospect we now realize that the design of the MYCIN study would have permitted several different kinds of patterns to be uncovered, any one of which could have been used as a basis for increasing the agreement among the evaluators, and hence the power of the test. Unfortunately, none of these patterns actually appeared in the MYCIN data.

### 30.3.3 Collapsing the Data

The previous discussion of the tendency of the experts to agree with one another is subject to at least one objection. Suppose that, for a particular case, four of the ten prescribers made the same recommendation, and

expert e1 agreed with the recommendation while expert e2 did not. Then e1 and e2 would be counted as disagreeing four times, when in fact they are only disagreeing over one question. If a large number of the cases lead to only a few different responses, then it might be worth lumping together the prescribers that made the same therapy recommendation. Then the experts will be interpreted as judging the responses the subjects made, rather than the subjects themselves. As is noted in the next section, this kind of collapsing of the data is useful for other purposes as well.

Deciding whether two treatments are identical may be nontrivial. Sometimes the responses are literally identical, but in other cases the responses will differ slightly, although not in ways that would lead a physician with a good understanding of the problem to accept one without also accepting the other. One plausible criterion is to lump together two therapy recommendations for a case if no evaluator accepts one without accepting the other. A second test is available when one of the evaluators gives a recommendation that is identical to one of the prescriber's recommendations. Recommendations that that evaluator judged to be equivalent to his own can then be grouped with the evaluator's recommendation, so long as doing so does not conflict with the first criterion. In using either of these tests, the data should first be made consistent in the manner discussed in Section 30.3.1.

Using these tests, the ten subjects in the ten cases of the MYCIN study reduced to an average of 4.2 different therapy recommendations for each case, with a standard deviation of 1.55 and a range from 2 to 6. This seems to be a large enough reduction to warrant looking at the data in this collapsed form.

### 30.3.4 Judges as Subjects

With the collapsing of prescribers into therapies, it may be possible to identify an evaluator's recommendation with one or more of the prescribers' recommendations. By then eliminating that evaluator from the rank of judges, his recommendation can be considered judged by the other evaluators. In this way the evaluators may be used as judges of each other, thereby allowing comparisons with the rankings of the original prescribers. This does not always work, since sometimes an evaluator's recommendation cannot be identified with any of the prescribers'. In Study 3, 9 out of 80 evaluator-generated therapies could not be judged as identical to any of the prescribers' recommendations.

Measuring the evaluators' performance against each other in this manner provides another indication of the extent of disagreement among them. It also produces more scores that can be (roughly) compared to the percentage scores of the prescribers. In Study 3, 8 more scores can be added to the 10 assigned to the prescribers, giving a field of 18 scores. The analysis of variance or chi-square was run on this extended population.

The new analysis showed that the mean score for the evaluators was

0.699, which is both higher than the mean agreement (0.591) and higher than the mean of the prescribers' scores (0.585). This latter fact is to be expected, since the subjects included people who were chosen for the study because their level of expertise was assumed to be lower than that of the evaluators. Nevertheless, half of the evaluators scored above the highest-scoring prescriber (while the other half spread out evenly over the range between the top-ranking subject and the eighth-ranking subject). The fact that agreement between the evaluators looks higher on this measure than it does on other measures indicates that much of the disagreement was over therapies that none of the evaluators themselves recommended.

It is interesting to ask why the evaluators ranked higher in this analysis than the Stanford faculty members among the prescribers, many of whom would have qualified as experts by the criteria we used to select the national panel. A plausible explanation is the method by which the evaluators were asked to indicate their own preferred treatment for each of the ten cases. As is described in Chapter 31, for each case the expert was asked to indicate a choice of treatment on the first page of the evaluation form and *then* to turn the page and rank the ten treatments that were recommended by the prescribers. There was no way to force the evaluators to make a commitment about therapy before turning the page, however. It is therefore quite possible that the list of prescribers' recommendations served as "memory joggers" or "filters" and accordingly influenced the evaluators' decisions regarding optimal therapy for some of the cases. Since none of the prescribers was aware of the decisions made by the other nine subjects, the Stanford faculty members did not benefit from this possible advantage. We suspect this may partly explain the apparent differences in ratings among the Stanford and non-Stanford experts.

### 30.3.5 Summary

The discussion in this section demonstrates many of the detailed subanalyses that may be performed on a rich data set such as that provided by Study 3. Information can be gathered on interscorer reliability of the evaluation instrument, and statistical techniques are available for detecting correlations and thereby increasing the reliability (and hence the power) of the test.