process: A tool for clinical supervision and the design of clinical information systems. *Journal of Medical Systems, 8*, 7–15.

Craig, T.J., Siegel, C. & Laska, E. (1982). Automation in clinical systems and quality assurance. *International Journal of Mental Health, 10*, 76–91.

Donabedian, A. (1966). Evaluating the quality of medical care. *Mulbank Memorial Fund Quarterly Supplement, 44*, 166–206.

Hammond, K. & Munnecke, T. (1984). A computerized psychiatric treatment planning system. *Hospital and Community Psychiatry, 35*, 160–163.

Hetherington, R.W. (1982). Quality assurance and organizational effectiveness in hospitals. *Health Services Research, 17*, 185–201.

Klerman, G.L. (1984). The advantages of DSM-III. *American Journal of Psychiatry, 141*, 539–542.

Laska, E. (1982). Developments in computerization of the psychiatric record. *International Journal of Mental Health, 10*, 54–75.

Minsky, M. (1981). A framework for representing knowledge. In J. Haugeland (Ed.) *Mind design: Philosophy, psychology and artificial intelligence*. Cambridge, MA: MIT Press.

Siegel, C. & Fischer, S.K. (Eds.) (1981). *Psychiatric records in mental health care*. New York: Brunner/Mazel.

Spitzer, R. & Endicott, J. (1974). Computer diagnoses in automated record keeping systems. In J.L. Crawford, D.W. Moyan & D.T. Gianturco (Eds.) *Progress in mental health information systems: Computer applications*. Cambridge, MA: Ballinger Publishing Company.

Tischler, G.L. (1974). Development of standards for evaluation in direct patient care. In D. Riedel, G. Tischler & J. Myers (Eds.) *Patient care evaluation in mental health programs*. Cambridge, MA: Ballinger Publishing Company.

Weed, L. (1968). Medical records that guide and teach. *New England Journal of Medicine, 278*, 593–600.

Computer Simulation of Human Service Program Evaluations

William M.K. Trochim
James E. Davis

KEYWORDS. Computer Simulation, Program Evaluation.

ABSTRACT. Computer simulations in human service research are useful for (1) improving student understanding of basic research principles and analytic techniques, and (2) investigating the effects of problems which arise in the implementation of research. This paper describes these uses of simulations for the context of human service program evaluation. Simple mathematical models are described for the three most commonly used human service outcome evaluation designs—the pretest-posttest randomized experiment, the pretest-posttest nonequivalent groups design, and the regression-discontinuity design. The models are translated into a single computer program which can be used to conduct the simulations, and examples of the use of this program are provided. The paper concludes that simulations need to utilize experimental design principles when rigorous, definitive results are desired, but that, even when this is not possible or desirable, simulations may have great potential value as an exploratory or teaching tool in human service research.

Imagine the teacher faced with the difficulties of explaining evaluation design to a class of human service students. The teacher has no problem in conveying the importance of defining the evaluation question, understanding the political context of the study, or involving different stakeholder groups in the research

William M.K. Trochim is Assistant Professor, Department of Human Service Studies, Cornell University, Ithaca, NY, 14853. James E. Davis is a Ph.D. candidate in the same department.

process. But when faced with the more formidable "technical" side of the evaluation process—the construction of measures, the choice of sampling plan, the selection of a research design, and the analysis of the data—the class becomes lost in the complexities of the material. How can the teacher convey the logic behind an Analysis of Covariance or a pretest-posttest nonequivalent group design in a way which is understandable to the students?

Or, imagine the human service evaluator who is in the process of supervising a program evaluation. A number of problems, partially or entirely unanticipated are beginning to arise. The evaluator is not sure whether all program participants are attending the program or even whether the program is being carried out in a similar way for all participants. Several of the measures for a small subgroup have been lost and the evaluator is having a hard time getting comparison group persons to come back to the agency for posttest measurement. In fact, the evaluator is not even very confident that the program and comparison groups were really "comparable" to begin with. How can this evaluator examine what the likely effects of some of these problems might be on the final results?

Computer simulation is a tool which can help [illegible] teacher and evaluator address these types of [illegible] simulation, the analyst first creates data according to a [illegible] and then examines how well the model can be detect[illegible] data analysis. The teacher can show student that mea[illegible]pling, design and analysis issues are dependent on the model which is assessed. Students can directly manipulate the simulation model and "try things out" to see immediately how results change and how analyses are affected. The evaluator can construct models of evaluation problems—making assumptions about the extent or kind of attrition, group nonequivalence or program implementation—and see whether the results of any data analyses are seriously distorted.

Simulations are better for some purposes than the analysis of "real" data. With "real" data, the analyst never perfectly knows the real-world processes which caused the particular observed values to occur. In a simulation, the analyst controls all of the factors which make up the data and can manipulate these systematically to see directly how specific problems and assumptions affect the analysis. Simulations also have some advantages over abstract theorizing about research issues. They enable the analyst to come into direct contact with the assumptions which are made and to develop a concrete "feel" for their implications on different analysis techniques.

Simulations have been widely used in contemporary social research (Guetzkow, 1962; Bradley, 1977; Heckman, 1981). They have been used in human service contexts, but to a much less degree (Mandeville, 1978; Raffeld, Stamman and Powell, 1979; Mandell and Blair, 1980). Most of this work has been confined to the more technical literatures in these fields.

The purpose of this paper is to explain and illustrate some basic principles of computer simulation and show how simulations may be used to improve the work of human service teachers and evaluators. The discussion will focus on a specific type of simulation context—the human service program or outcome evaluation. The paper will describe the three most commonly used human service program evaluation designs, present a microcomputer program for simulating these designs, and discuss the use of this program in human service teaching and the study of evaluation implementation.

THE SIMULATION MODELS

In the human services we are often interested in evaluating the effects of outcomes of some programs. In education, the program might consist of compensatory training in mathematics for grade school children. In criminal justice it might be a novel diversion program for first-time offenders. In mental health, it might be a unique combination of services designed to aid recently deinstitutionalized clients.

Whatever the program, the evaluator will typically need to select one of three major strategies or designs for conducting the evaluation. All three designs, in their simplest forms, involve pre and post-program measurement of both program and comparison group participants. The three designs differ in the way in which persons are assigned to participate in the program. In the randomized experimental (RE) design, persons are randomly assigned to either the program or comparison group. In the regression-discontinuity (RD) design (Trochim, 1984), all persons who score on one side of a chosen preprogram measure cutoff value are assigned to one group, with the remaining persons being assigned to the other. In the nonequivalent group design (NEGD) (Cook and Campbell, 1979; Reichardt, 1979), persons or intact groups (e.g., classes, wards, jails) are "arbitrarily" assigned to either the program or comparison condition. These designs have been used extensively in human service evaluations where one is interested in determining whether the program had an effect on one or more outcome measures. The technical literature on these designs is extensive (see, for instance, Cook and Campbell, 1979; Trochim, 1986) and a discussion of their relative advantages is outside the scope of this paper. The general wisdom is that if one is interested in establishing a *causal* relationship (e.g., in internal validity), RE designs are most preferred, the RD design (because of its clear assignment-by-cutoff rule) is next in order of preference, and the NEGD is least preferable.

All three of the program evaluation designs (RE, RD and NEGD) have a similar structure which can be described as follows:

Program Group: Pretest ⟶ Program ⟶ Posttest

Comparison Group: Pretest ⟶ Posttest

The program group and comparison group are represented on separate lines and passage of time is indicated by movement from left to right in the diagram. Thus, the program group is given a pre-program measure (often termed the "pretest"), is then given the program, and afterward is given the post-program measure (posttest). The vertical similarity in the measurement structure implies that both the pre and post measures are given to both groups at the same time. To simulate the designs, we begin by constructing a model for each one. The model specifies the structure of the pretest, the strategy for assigning persons to program or comparison group, the size of the program effect, and the structure of the posttest. In the following sections we provide a simple model for each of the three designs.

The RE Model

To construct the model for the RE design we begin with the assumption that the preprogram measure, X, is the additive function of two components—a true score, t, and a random error factor e_x, such that

$$x = t + e_x$$

For each case (or hypothetical person) we randomly generate both t and e_x and add these together to create the pretest. Next, a variable, z, which describes group membership (i.e., program or comparison) is constructed such that

$$\begin{aligned} z &= 1 \text{ if } r \le 0 \\ &= 0 \text{ otherwise} \end{aligned}$$

where

z is a (0,1) dummy-coded assignment variable
r is a normal random variable and is independent of all other terms

To accompish this, we simply generate for each new case a new random variable, r, which is normally distributed with a mean equal to 0 and some standard deviation. Then the case is assigned to the program (z = 1) or comparison (z = 0) group according to the above rule. Finally, we construct the post-program measure, y, such that for each case

$$y = t + e_y + (gz)$$

where

y is the post-program measure
t is the same true score as used for the pretest
e_y is a normal random variable and is independent of all other terms
g is the program effect size
z is group membership as defined above

For each case, the post-measure is an additive composite of the same true ability (t) as in the pre-measure, an independent error (e_y) and an effect size (gz). It is important to note that the effect (g) is only added to program group cases when comparison group cases z = 0 and the product gz therefore also equals 0.

The NEGD Model

In the nonequivalent group design we assign persons or units to conditions *nonrandomly*. As a result, we expect that the two groups may differ systematically in ability as reflected in the measures. If, for instance, two classrooms or hospital units are arbitrarily assigned to receive the program or not, it is plausible to assume that the two groups will, on average, differ to some degree on both the pre and post measure, even in the absence of the program. In simulations, we can deliberately create such nonequivalence by adding some constant value to both the pre and post measure for one of the groups. Therefore, to construct the model for this design we first need to create the group assignment variable, z, in the same way as for the RE design

$$\begin{aligned} z &= 1 \text{ if } r < 0 \\ &= 0 \text{ otherwise} \end{aligned}$$

where r is a normal random variable as defined before. Once this is accomplished, we can create the pre-program measure, x, such that for each case

$$x = t + e_x + (dz)$$

where t is a true score and e_x is a random error factor. Here, d is a constant which is added to the program group (note that it can be either positive or negative depending upon whether one wishes the program group to be "advantaged" or "disadvantaged," relative to the comparison cases) through multiplication with the (0,1) dummy-coded group assignment variable. The post-measure, y, is constructed for each case, such that

$$\begin{aligned} y &= t + e_y + (dz) + (gz) \\ &= t + e_y + z(d + g) \end{aligned}$$

where

y is the post-program measure
t is the same true score as for the pretest
e_y is a normal random variable and is independent of all other terms
d is a constant representing group nonequivalence as used for the pretest

g is the program effect size
z is the (0,1) group membership indicator

The RD Model

The model for the RD design can be constructed by beginning with the pre-measure, x, such that for each case

$$x = t + e_x$$

where the pre-program measure x is again the additive function of a true score, t, and a random error factor, e_x. Next, the group membership variable, z, can be constructed for each case such that

$$\begin{aligned} z &= 1 \text{ if } x \leq \text{(cutoff value)} \\ &= 0 \text{ otherwise} \end{aligned}$$

There are two important points to note. First, one must select a cutoff value on the pre-measure. Second, the RD design requires that either low or high scorers be assigned to the program group depending on the nature of the evaluation. If the program involves special training in mathematics which should be given to "needy" students and the pre-measure is an indicator of prior math ability (where low scores indicate poor math performance) then all students scoring *below* some pre-measure cutoff value would be given the program (as in the above formula which would be appropriate for this "compensatory" situation). However, if the program involves a novel surgical technique which should only be piloted on the most needy cases and the pre-measure is an indicator of the severity of illness (where *high* scores indicate the *greatest* need) persons with pre-measure scores *above* some value would be assigned to the program and the formula above would need to be adjusted accordingly.

Finally, the post-program measure, y, is constructed for each case such that

$$y = t + e_y + (gz)$$

which is an identical formula to the one used for the RE design but differs significantly in that the definition of z, the group membership indicator, is a cutoff-based rather than random assignment indicator.

Summary of Model Building Procedures

All three designs have the same structure in that for each a pre-program, post-program, and treatment (dummy variable) measure is created. However, the models presented above show that the designs differ considerably in how these three terms are constructed. In the next section these three models will be translated into a single computer program which will be used for simulations.

THE SIMULATION PROGRAM

The models for the three designs can be efficiently simulated with a single computer program. This is illustrated here with a program written in the MINITAB statistical computing system (Ryan, Joiner and Ryan, 1982)[1] shown in Appendix A. The first section of the program involves the specification of six constants (K1–K6) which define the parameters for the simulation. By changing the values of these constants, one can alter the size of the program effect, the degree of nonequivalence (in the NEGD), the reliability of the measures, and the sample size. The random variables which are needed for all three models (t, e_x, e_y and r) are generated in four "nran" statements. The next few sections on the listing describe the construction of the x, y, and z variables for the three models. Note that in the sample program a premeasure cutoff value of 0 was chosen for the RD design. Table 1 lists the MINITAB variables and variable names which correspond to the pretest (x), group assignment (z) and posttest (y) for the three models. For each model, the program then prints out the group means and standard deviations. Next, bivariate plots are constructed for each model (on output, the letter A indicates a program case; the letter Z indicates a comparison one; a number indicates the number of cases which fall on the same point; and, an * indicates that more than nine cases fall on the same spot). All three designs are analyzed using the same Analysis of Covariance (ANCOVA) regression model:[2]

$$Y_i = b_0 + b_1x_i + b_2z_i + e_i$$

where

Y = posttest score for case (i.e., person) i
b_0 = constant or intercept parameter

b_1 = linear slope of y on x parameter
x_i = pretest score for case i
b_2 = program effect parameter
z_i = group assignment for case i
e_i = residual for case i

In each analysis, the three estimated parameters, b_0, b_1, and b_2, are saved and the key estimate, b_2, the estimate of the program effect, is stored in a new variable. These results are cumulated over successive runs of the simulation.

The program can be executed interactively by typing each command as presented in Appendix A (note that commands which begin with # are comments and need not be typed. Alternatively, the program can be stored in a standard system file and executed n times using the MINITAB command

```
execute 'filename' n
```

All examples presented here were run on an IBM PC/XT microcomputer equipped with an 8087 math co-processor chip.

Table 1
Index of MINITAB variables and variable names for the RE, NEGD and RD Models

	RE	NEGD	RD
	Variable and Name	Variable and Name	Variable and Name
pretest (x)	C5	C10	C5
	x^1-RE-RD	x-NE	x-RE-RD
group (z)	C6	C6	C8
	z-RE-NE	z-RE-NE	z-RD
posttest (y)	C7	C11	C9
	y-RE	y-NE	y-RD

[1] This is the x variable for the randomized experiment and regression discontinuity designs.

SIMULATION APPLICATIONS AND VARIATIONS

There are a number of ways in which the simulations described here (and simple variations of the program given above) can be useful in human service program evaluation contexts. First, they provide a powerful teaching tool (Eamon, 1980, Lehman, 1980). Students of human service program evaluation can explore the relative advantages of these designs under a wide variety of conditions. In addition, the simulations show the student exactly how an analysis of these designs could be accomplished using real data. Second, the simulations provide a way to examine the possible effects of evaluation implementation problems on estimates of program effect (Trochim and Spiegelman, 1980; Trochim, 1982; Muthen and Joreskog, 1984). Just as NASA explores difficulties in a space shuttle flight using an on-ground shuttle simulator, the data analyst can examine the possible effects of attrition rates, floor or ceiling measurement patterns, and other implementation factors on the size of the program effect.

Applications for Teaching

To illustrate the utility of the simulation program for teaching, two example simulations were run. The first shows how the three designs perform when measurement is highly reliable while the second illustrates what happens when the measures are low in reliability.

For the "high reliability" example, the program effect for all models was 5 points, the NEGD program group had a three-point "advantage" (i.e., was nonequivalent and advantaged on pre and post measures), the pre-measure cutoff value was zero for the RD design, the reliability of the measures was equal to .9 (see below) and there were 100 cases in each of the 50 runs. In the "low reliability" example, all simulation parameters remained the same except that the reliability of the measures was .5, considerably lower than before.

The reliability of the measures was set by varying the relative size of the standard deviations of the true and error scores. Reliability is defined as

$$\text{rel} = \frac{\text{var (t)}}{\text{var(t)} + \text{var(e)}}$$

Therefore, if we set K3 in the program (standard deviation of the true scores) equal to 3 and K4 (standard deviation of the error scores) equal to 1, we obtain the reliability

$$\begin{aligned} \text{rel} &= \frac{3^2}{3^2 + 1^2} \\ &= \frac{9}{10} \\ &= .9 \end{aligned}$$

for the first "high reliability" simulations. In the second "low reliability" simulations, we set K3 = 3 and K4 = 3 and thereby obtain

$$\begin{aligned} \text{rel} &= \frac{3^2}{3^2 + 3^2} \\ &= \frac{9}{18} \\ &= .5 \end{aligned}$$

for the reliability of the measures.

The cumulative results for 50 runs for these two simulation examples are shown in Table 2.The results illustrate some important methodological principles. First, both the RE and RD designs yield unbiased estimates. In general, we would consider estimates to be unbiased if the average gain does not differ positively or negatively by more than two standard error units from the true gain (i.e., a .05 significance level where the gain, g, falls within the interval $b_2 \pm 2SE_{b2}$). For instance, for the RE design, low reliability simulations, the average gain is 4.89 and the standard error is .081. Therefore, the true gain, 5 points, falls within the interval 4.89 ± 2(.081) and the RE design can be considered unbiased for these conditions. Second, the NEGD is shown to yield biased estimates of effect for both low and high reliability simulations. This is consistent with the literature on this design (Reichardt, 1979), which maintains that the ANCOVA analysis will yield biased estimates of effect when the pretest is not perfectly measured (i.e., there is measurement error on the pretest). Finally, the results show that the

Table 2

Simulation Results for the Basic Program

(true gain=5.0, 50 runs, n=100 per run)

		b_2	SE (b_2)	min (b_2)	max (b_2)
High Relia-bility (.9)	RE	4.973	.041	4.251	5.578
	NEGD	5.252	.044*	4.448	5.884
	RD	5.032	.066	3.702	5.951
Low Relia-bility (.5)	RE	4.890	.081	3.663	6.401
	NEGD	6.344	.094*	4.521	7.930
	RD	5.030	.180	2.280	7.870

* Significance of coefficient is determined by its value falling outside of the range of 2 standard errors ($b_2 \pm 2SE_{b2}$).

designs differ in efficiency. For both the high and low reliability simulations the RE and NEGD have similar stan[illegible] the average gain, whereas the RD design standard e[illegible] ably larger. This also is consistent with the litera[illegible] (1972), for instance, demonstrated that, all things be[illegible] the RD design requires 2.75 times the number of cases [illegible] design in order to have the same relative efficiency.

How can simulations of this type be useful for teaching about human service program evaluations? First, students can observe the simulation program as it is executing on the computer and get an idea of how a real data analysis might look. In addition, the simulation presents the same information in a number of ways. The student can come to a better understanding of the relationships between within-group pretest and posttest means and standard deviations, bivariate plots of pre and post measures which also depict group membership, and the results of the ANCOVA regression analyses. Second, the simulations clearly demonstrate the probabilistic foundations of hypothesis testing in this context. For instance, the results shown in Table 2 illustrate that even with measures which are fairly reliable, one will sometimes obtain estimates of effect which are near the true value (even when the analysis yields biased results on average, as with the NEGD) or estimates which differ considerably from the true value (even when the analysis yields unbiased estimates on average). To demonstrate these notions even more directly, the student can display for each design the histograms of the estimates of effect across a number of simulation runs. Third, the simulations illustrate clearly some of the key assumptions which are made in these designs and allow the student to examine what would happen if these assumptions are violated. For instance, the simulations are based on the assumption that within-group pre-post slopes are linear and that the slopes are equal between groups. The effects of allowing the true models to have treatment interaction terms or nonlinear relationships can be examined directly with small modifications to the simulation program as Trochim (1984) illustrated for the RD design. Fourth, the simulations demonstrate the importance of reliable measurement. By varying the ratio of true score and error term variances, the student can directly manipulate reliability and show that estimates of effect become less efficient as measures become less reliable. Finally, simulations are an excellent way to illustrate that apparently sensible analytic procedures can yield biased estimates under certain conditions. This is shown most clearly in the simulations reported in Table 2 for the NEGD. While the apparent similarity between design structures of the RE and NEGD might suggest that traditional ANCOVA regression models are appropriate, the simulations clearly show this to be false and thereby confirm the statistical literature in this area (Reichardt, 1979).

Applications for the Study of Design Implementation

The validity of estimates from the three designs described here depends on how well they are executed or implemented in the field. There are many implementation problems which occur in typical human service program evaluations—attrition problems, data coding errors, floor and ceiling effects on measures, poor program implementation, and so on—which degrade the theoretical quality of these designs (Trochim, 1984). Clearly, there is a need for improved evaluation quality control (Trochim and Visco, 1985), but when implementation problems cannot be contained, it is important for the analyst to examine the potential effects of such problems on estimates of program gains. This application of simulations is analogous to simulation studies which NASA conducts to try to determine the effects of problems in the functioning of the space shuttle or a communications satellite. There, an exact duplicate of the shuttle or satellite is used to try to recreate the problem and explore potential solutions. In a similar way, the program evaluator can attempt to recreate attrition patterns or measurement difficulties to examine their effects on the analysis and attempt to discover analytic corrections which may minimize these effects.

The simulation program is illustrated using examples of two common evaluation implementation problems. The first example looks at what happens to estimates of program effect when there is attrition from the study. The second example examines measurement ceiling effects and their consequences.

For our first example, the modelling of attrition patterns, we need to make assumptions about what causes attrition in the context at hand. Here, we will make a rather simple assumption for purposes of illustration: that persons (or cases) who are low in *true ability* on pre and post measures are the most likely attrition cases. This might be the case in educational contexts where it may be the lowest ability students who are lacking motivation or are erratic in attendance and therefore are excluded from the data analysis for want of either a pre or post program score. Similarly, in health or mental health contexts it may be the most needy or the most severely ill who contribute most to the attrition rate. We can operationalize this attrition assumption in a somewhat crude way by excluding all cases in the simulation which have true scores (i.e., true ability) lower than some chosen value. In these simulations, the attrition model was accomplished with the addition of the following program statement immediately after the random generation of the true scores:

```
recode −100 −1.5 C1 '*' C1
```

This command assigns the MINITAB missing value code to all the cases that have a true score lower than −1.5. The selection of −1.5 is arbitrary here and was chosen to allow enough attrition to be detectable in this example. These cases are subsequently removed from the analysis. As in the previous example, all three models were simulated for both low and high reliability measurement. The average estimates of effect, standard errors and minimum and maximum estimates are shown in Table 3.

The results suggest several lessons. As in the previous simulation, the RE design appears to yield unbiased estimates for both high and low reliability conditions. Although the attrition pattern is systematic with respect to true ability (and is therefore correlated with both the pre and post measures) it is random with respect to the assignment variable, r. The NEGD clearly yields biased estimates, and these are even more biased than in the previous non-attrition simulations. The RD design is clearly biased under the high reliability model and is marginally biased for the low reliability condition. This suggests that a greater number of simulation runs (or a larger n for each run) might indicate that the RD design generally yields biased estimates under this attrition model. This example clearly shows the advantage of the RE design when attrition is correlated with true ability and is not differential between groups.

Table 3

Simulation Results for the Attrition Model

(true gain=5.0, 50 runs, n=100 per run before attrition)

		b_2	SE (b_2)	min (b_2)	max (b_2)
High Relia-bility (.9)	RE	5.074	.052	4.400	5.873
	NEGD	5.608	.058*	4.713	6.610
	RD	5.202	.074*	3.902	6.654
Low Relia-bility (.5)	RE	4.899	.130	2.819	7.057
	NEGD	6.888	.125*	4.636	8.821
	RD	5.310	.180	2.990	7.970

* Significance of coefficient is determined by its value falling outside of the range of 2 standard errors ($b_2 \pm 2SE_{b2}$).

The second example of the use of simulations for investigating implementing problems involves the construction of a ceiling effect on the post-program measures for all three designs. A ceiling effect occurs when a measure is unable to discriminate between the ability levels of persons who do well on the test. When a test is too easy, for instance, many respondents may achieve perfect or near-perfect scores. The scores cannot be considered accurate indicators of their relative ability because, if the test were harder, some respondents would outscore others at this upper level. The problem is especially troubling when it occurs on a post-program measure which is presumed to reflect program-related gains. Instead, potential gains will be masked by the test's inability to allow higher posttest scores.

A simple model for constructing a posttest ceiling effect was constructed in these simulations by forcing all program cases having a 6.5 or greater on the posttest to be given the posttest ceiling value of 6.5 instead.[3] This is easily accomplished by inserting the following three statements immediately before naming the variables in the program:

```
reco  6.5  100  C7   6.5  C7
reco  6.5  100  C9   6.5  C9
reco  6.5  100  C11  6.5  C11
```

In MINITAB, the recode command can also be stated *reco*. The ceiling effect value of 6.5 is arbitrary here and was chosen for illustrative purposes. The average estimates of effect, standard errors, and minimum and maximum estimates for both the high and low reliability conditions are shown in Table 4 for the three designs.

In this example, all three models yield biased estimates of effect for both high and low reliability conditions. In all cases but one, the bias is in the direction of *underestimating* the true effect. This is not surprising given that there was a posttest ceiling which prevented larger gains from occurring. In the only exception, the RD design under the high reliability condition, the effect is *overestimated* due to the nature of the regression model which is used. A more detailed consideration of this result is outside the scope of this paper and the reader is referred to Trochim (1984) for a more extensive discussion of the RD design and the analytic problems which can lead to this pattern of results.

Table 4

Simulation Results for Posttest Ceiling Effect

(true gain=5.0, 50 runs, n=100 per run)

		b_2	SE (b_2)	min (b_2)	max (b_2)
High Relia-bility (.9)	RE	4.242	.036*	3.692	4.959
	NEGD	3.911	.041*	3.465	4.635
	RD	5.218	.068*	4.014	5.987
Low Relia-bility (.5)	RE	3.801	.084*	2.477	5.310
	NEGD	4.267	.083*	3.056	5.686
	RD	4.050	.150*	0.950	6.310

* Significance of coefficient is determined by its value falling outside of the range of 2 standard errors ($b_2 \pm 2SE_{b2}$)

The attrition and posttest ceiling examples illustrate the use of simulations to examine common research implementation problems. The analyst can directly manipulate the parameters in the models (e.g., using different values for creating attrition or ceiling effects) in order to approximate their reality more accurately and to examine the performance of the design under more varied situations. Such simulations are useful in that they can alert the analyst to potential bias and even indicate the direction of bias under the various assumptions.

DISCUSSION

This paper describes several simple simulation models which are appropriate for a few, relatively confined, human service situations, namely, the use of three common research designs for evaluating human service program effects. Nevertheless, the logic of these simulations is easily extended to other relevant human research contexts. For instance, many human service agencies routinely conduct sample surveys to identify needs and target populations, assess services which are provided or, compare agency functioning with the performance of other similar agencies or with some standard. One would construct simulation models for survey instruments for the same reasons that they are constructed for evaluation designs—to improve teaching and general understanding and to explore problems in implementing the survey (e.g., non-response patterns). The key to doing this would again rest on generating statistical models which are used to create hypothetical survey responses. A "true score" measurement model is useful here, at least for simple simulations, but may have to be modified. For instance, assume that one question on a survey deals with client satisfaction with a particular service and that the response is a 7-point Likert-type format where 1 = very dissatisfied, 7 = very satisfied, and 4 = neutral. The analyst could make the assumption that for some sample or subsample the true average response is a scale value equal to 5 points (somewhat satisfied), and that the true distribution of responses is normal around this value, with some standard deviation. At some point, the analyst will have to convert this hypothetical underlying continuous true distribution to the

7-point integer response format either by rounding or by generating normally-distributed random integers in the first place. Such a variable could then be correlated or cross-tabulated with other generated responses to explore analytic strategies for that survey. Similar extensions of the models discussed here can be made for simulations of routinely-collected management information system (MIS) information, for data for correlational studies, or for time-series situations, among others.

Simulations are assumptive in nature and vary in quality to the degree that the reality is correctly modelled. When constructing a simulation, it is important that the analyst seek out empirical evidence to support the assumptions which are made, whenever this is feasible. For instance, it should be clear that the simulations described here could be greatly enhanced if we had more specific data on how much and what type of attrition typically occurs, what type of floor or ceiling effects are common, what patterns of misassignment relative to the cutoff value typically arise for the RD design, and so on. While some relevant data will be available in the methodological literature, all of these issues are context specific and demand that the analyst know the setting in some detail if the simulations are to be reasonable.

One way to approach the assumptive nature of the simulation task is to recognize that reality conditions or parameters in the models need to be examined systematically across a range of plausible conditions. This implies that multiple analyses under systematically varied conditions which are based upon principles of parametric experimental design are needed in "state-of-the-art" simulation work. This point is made well by Heiberger et al. (1983) who state:

> The computer has become a source of experimental data for modern statisticians much as the farm field was to the developers of experimental design. However, many "field" experiments have largely ignored fundamental principles of experimental design by failing to identify factors clearly and to control them independently. When some aspects of test problems were varied, others usually changed as well—often in unpredictable ways. Other computer-based experiments have been ad hoc collections of anecdotal results at sample points selected with little or no design. (p. 585)

Heiberger et al. (1983) go on to describe a general model for simulation design which allows the analyst to control systematically a large number of relevant parameters across some multidimensional reality space, including the sample size, number of endogenous and exogenous variables, number of "key points" or condition values, intercorrelations, least squares regression coefficients, means, standard errors, and so on. Although simulations play an essential role in such rigorous investigation of statistical procedures, they may be even more important for the human services in that they enable the evaluator to explore simple relevant data structures in order to improve teaching about human service research and to explore potential research implementation problems.

ENDNOTES

1. Most commonly available statistical packages could be used. Analogous program listings for SPSS/X and SAS are available upon request from the first author. The program presented in this paper is machine independent and will run on any comparable version of MINITAB. The MINITAB version is presented here because that language is widely available on micros, minicomputers and mainframes; is relatively inexpensive; and, is easy to learn.

2. ANCOVA models are described in greater detail in Pedhazur (1982), Myers (1972) and Keppel (1973). The analysis of ANCOVA models using standard regression analysis computer programs is described in Nie et al. (1975).

3. In order to make the posttest ceiling conditions similar across the three designs it was necessary to alter the assignment procedure for the RD design so that the program group consisted of cases scoring *above* the cutoff value rather than below it. This is accomplished by replacing the three statements used to create the RD assignment measure, c8, with the following:

```
reco  -100 0 c5 -1 c8
reco  0 100 c8 1 c8
reco  -1 c8 0 c8
```

Thus, all cases having a pretest greater than zero are in the program group and all remaining cases are in the comparison group. This variation might arise in practice if the program is given to "advantaged" persons (e.g., a scholarship or award) or if the pre-measure is an indicator of need where high scores indicate greater need.

REFERENCES

Bradley, D. R. (1977). Monte Carlo simulations and the chi-square test of independence. *Behavior Research Methods and Instrumentation*, 9, 193–201.

Campbell, D. T., & Boruch, R. F. (1975). Making the case for randomized assignment of treatments by considering the alternatives: Six ways in which quasi-experimental evaluations tend to underestimate effects. In C. A. Bennet, & A. A. Lumsdaine (Eds.)

Evaluation and experience: Some critical issues in assessing social programs. New York: Academic Press.

Cook, T. D., & Campbell, D. (1979). *Quasi-experimentation: Design and analysis issues for field settings.* Chicago: Rand McNally.

Eamon, D. E. (1980). Labsim: A data-driven simulation program for instruction in research design and statistics. *Behavior Research Methods and Instrumentation.* 12, 160–164.

Goldberger, A. S. (1972). *Selection bias in evaluating treatment effects. Some formal illustrations.* Discussion Papers, 123-72. Madison: Institute for Research on Poverty, University of Wisconsin.

Guetzkow, H. (1962). *Simulation in social science.* Englewood Cliffs, N.J.: Prentice-Hall, Inc.

Heckman, J. (1981). The incidental parameters problem and the initial conditions in estimating a discrete time-discrete data stochastic process. In C. F. Manski, & D. McFadden (Eds.), *Structural analysis of discrete data with economic applications.* Cambridge, Mass.: MIT Press.

Heiberger, R. M., Velleman, P. F., & Ypelaar, A. M. (1983). Generating test data with independent controllable features for multivariate general linear forms. *Journal of the American Statistical Association,* 78, 585-595.

Keppel, G. (1973). *Design and Analysis: A Researcher's Handbook.* Englewood Cliffs, NJ: Prentice-Hall, Inc.

Lehman, R. S. (1980). What simulations can do to the statistics and design course. *Behavior Research Methods and Instrumentation,* 12, 157-159.

Mandell, L. M., & Blair, E. L. (1980). Forecasting and evaluating human service system performance through computer simulation. *American Statistical Association Proceeding,* Social statistics section, 60–67.

Mandeville, G. K. (1978). An evaluation of Title I model cl: The special regression model. *American Education Research Association Proceedings.*

Muthen, B., & Joreskog, K. G. (1984). Selectivity problems in quasi-experimental studies. In R. F. Conner, D. G. Altman, & C. Jackson (Eds.), *Evaluation Studies Review Annual,* Beverly Hills: Sage, Vol. 9.

Myers, J. L. (1972). *Fundamentals of Experimental Design* 2nd Edition. Boston: Allyn & Bacon.

Nie, N. H., Hull, C. H., Jenkins, J. G., Steinbrenner, K., Bent, D. H. (1975). *SPSS: Statistical Package for the Social Sciences* 2nd Edition. NY: McGraw Hill

Pedhauzur, E. J. (1982). *Multiple Regression in Behavioral Research* 2nd Edition. NY: Holt, Rhinehart & Winston.

Raffeld, P., Stamman, D., & Powell, G. (1979). A simulation study of the effectiveness of two estimates of regression in the Title I model A procedure. *American Educational Research Association Proceedings.*

Reichardt, C. (1979). The design and analysis of the non-equivalent group quasi-experiment. Unpublished doctoral dissertation.

Ryan, T. A., Joiner, B. L., & Ryan, B. F. (1982). *Minitab student handbook.* North Scituate, MA: Duxbury Press.

Trochim, W. (1982). Methodologically based discrepancies in compensatory education evaluation. *Evaluation Review,* 6, 3, 443–480, August. Reprinted in Light, R. J. (Ed.) *Evaluation Studies Review Annual, Volume 8.* Beverly Hills, CA: Sage Publications, 1983, 633-670.

Trochim, W. (1984). *Research design for program evaluation: The regression discontinuity approach.* Beverly Hills, CA: Sage.

Trochim, W. (1986, in press). Advances in quasi-experimental design and analysis. *New directions for program evaluation.* San Francisco, CA: Jossey-Bass, Inc.

Trochim, W., & Spiegelman, C. H. (1980). The relative assignment variable approach to selection bias in pretest-posttest group designs. *Proceeding of the Social Statistics Section,* American Statistical Association.

Trochim, W., & Visco, R. (1985). Quality control in evaluation. In D. S. Cordray (Ed.), Utilizing prior research in evaluation planning. *New Directions in Program Evaluation,* San Francisco, CA: Jossey-Bass Inc., No. 27, September.

Appendix A

MINITAB Program to Simulate Three Human Service Program Evaluation Designs

```
# MINITAB Program to simulate a simple pretest-posttest
#    randomized experiment (RE), nonequivalent group (NE) design,
#    and regression-discontinuity (RD) design.
#
# Define simulation parameters
#
let k1=5   # k1 is the gain or program effect
let k2=3   # k2 is the selection bias for the NEGD
let k3=0   # k3 is the mean of the true scores
let k4=3   # k4 is the standard deviation of the true scores
let k5=1   # k5 is the standard deviation of the error terms
let k6=100 # k6 is the number of cases desired
#
# Set MINITAB environment parameters
#
batch
noprint
brief
#
# Generate random variables needed
#
nran k6 k3 k4 c1   # generate true score
nran k6 0 k5 c2    # generate pretest error
nran k6 0 k5 c3    # generate posttest error
nran k6 0 k5 c4    # generate assignment error
#
# Construct pretest for RE and RD
#
let c5=c1+c2 # pretest score
#
# Construct z and y for RE
#
reco -100 0 c4 -1 c6
reco 0 100 c6 0 c6
reco -1 c6 1 c6
let c7=c1 + (k1*c6) + c3
#
# Construct z and y for RD
#
reco -100 0 c5 -1 c8
reco 0 100 c8 0 c8
reco -1 c8 1 c8
let c9=c1 + (k1*c8) + c3
#
# Construct x and y for NEGD
#
let c10=c1 + c2 + (k2*c6)
let c11=c1 + c3 + ((k1+k2)*c6)
#
# Name the variables
#
name c1='true' c2='x-error' c3='y-error' c4='a-error'
name c5='x-RE-RD' c6='z-RE-NE' c7='y-RE' c8='z-RD'
name c9='y-RD' c10='x-NE' c11='y-NE'
```

Appendix A, continued

```
#
# Group statistics for randomized experiment
#
table c6;
stats c5 c7.
#
# Group statistics for nonequivalent group design
#
table c6;
stats c10 c11.
#
# Group statistics for regression-discontinuity design
#
table c8;
stats c5 c11.
#
# Bivariate plot for randomized experiment
#
lplot c7 c5 c6
#
# Bivariate plot for nonequivalent group design
#
lplot c11 c10 c6
#
# Bivariate plot for regression-discontinuity design
#
lplot c9 c5 c8
#
# Regression analysis for randomized exp
#
regr c7 2 c5 c6 c20 c21 c22
pick 3 3 c22 c23
join c23 c31 c31
#
# Regression analysis for nonequivalent g
#
regr c11 2 c10 c6 c20 c21 c22
pick 3 3 c22 c23
join c23 c32 c32
#
# Regression analysis for regression-discontinuity design
#
regr c9 2 c5 c8 c20 c21 c22
pick 3 3 c22 c23
join c23 c33 c33
#
# Name results variables and display aggregate results
#
name c31='REresult' c32='NEresult' c33='RDresult'
desc c31 c32 c33
```

Computer Work Skills Training for Persons With Developmental Disabilities

Thomas T. Saka

KEYWORDS. Computer Skills, Developmentally Disabled.

ABSTRACT. Research has shown that persons with Developmental Disabilities (DD) can hold a wide range of jobs, given the proper training and placement. The computer, the technological tool of the decade, has been used successfully in the education of persons with handicaps. Five subjects with DD were selected for this six month study and trained to use the microcomputer in performing basic data entry and word processing tasks. Four of the five subjects were placed in computer-related occupations following the training period.

In the past decade the use of computers in the work place and the home has increased at an astounding rate. To businesses, the computer is the latest innovation in an ever changing technological world. It is also a means for persons with handicaps to hold jobs and attain a level of learning previously thought to be unattainable.

Special education programs throughout the country have begun to use micro computers to increase the learning and communication abilities of students. For example, Brady and Bill (1984), Stallard (1982), and Schiffman, Tobin, and Buchanan (1982) demonstrated success with computer-assisted learning in regular education classrooms. Special education is an area which may be able to gain the most from these new teaching aids. Studies of the use of computers

This study was sponsored by the Hawaii State Planning Council on Developmental Disabilities. Thomas T. Saka is a graduate student in the Department of Educational Psychology at the University of Hawaii. He obtained his M.Ed. at the University of Hawaii. Requests for reprints should be sent to Thomas Saka who is at Department of Educational Psychology, University of Hawaii, 1776 University Ave., Honolulu, HI 96822.