SOME MEASUREMENT PROBLEMS DETECTED IN THE ANALYSIS OF SOFTWARE PRODUCTIVITY DATA AND THEIR STATISTICAL CONSEQUENCES

Jorge L. Romeu
Shirley A. Gloss-Soler

Reprinted from PROCEEDINGS OF THE IEEE COMPUTER SOCIETY'S INTERNATIONAL COMPUTER SOFTWARE AND APPLICATIONS CONFERENCE (COMPSAC), November 7-11, 1983

IEEE COMPUTER SOCIETY REPRINT

IEEE COMPUTER SOCIETY
1109 Spring Street, Suite 300,
Silver Spring, MD 20910

IEEE
COMPUTER
SOCIETY
PRESS

THE INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS, INC.

SOME MEASUREMENT PROBLEMS DETECTED IN THE ANALYSIS OF SOFTWARE PRODUCTIVITY DATA AND THEIR STATISTICAL CONSEQUENCES

Jorge L. Romeu and Shirley A. Gloss-Soler

Data & Analysis Center for Software*
IIT Research Institute
199 Liberty Plaza, Rome, New York 13440

Abstract

The Data and Analysis Center for Software (DACS) has been conducting an analysis of its productivity data. This analysis reveals that the level of the measurement scale of the productivity data is not high enough for the appropriate use of parametric statistical methods. Non-parametric equivalent methods are proposed as a valid alternative. An overview of the possible causes of this measurement weakness and some statistical consequences of analyzing the productivity data through both parametric and non-parametric procedures are presented. Finally, the practical importance of the study and classification of the data, including their measurement scale level, and the importance of selecting statistical methods that account for the data measurement problems are highlighted through a real-life numerical example.

1.0 INTRODUCTION

While analyzing the possible effects of Modern Programming Practices (MPP) on development efforts using the DACS Productivity[1] and NASA/SEL[2] datasets, it was found that several technical anomalies kept surfacing time and again. These types of anomalies posed great constraints on the statistical results. Attention was turned then to finding the possible causes of these problems, i.e., to the data being analyzed. Focusing on the procedures employed in collecting these data and on the procedures employed in the software development activity itself, it was concluded that a possible cause could be lack of the necessary strength in the measurement scale level in which the data were given. In other words, the statistical methods which had been generally employed in the analysis of the productivity data required a level from the data measurement scale which was higher than the actual measurement scale level in which the data was given. Most of the statistical problems stemmed from this measurement problem.

This paper presents an overview of the problem of data measurement scale and of the diagnosed particular situation for the present case and proposes a solution in the form of alternative statistical procedures defined to deal with these kinds of problems. With the presentation of this real-life data analysis, two important and preliminary analysis activities are underlined:

i) The proper study and classification of the data, including their measurement scale, as a factor in determining the analysis approach

ii) The differences in the assumptions for, the implications of selecting between parametric and non-parametric approaches, and where each is appropriate

Finally, for those interested in a more detailed treatment of this subject, a complete development can be found in (ROME82b).

2.0 METHODS

2.1 Background

Grossly speaking, a measurement scale is the kind of yardstick with which we determine a given characteristic. It can attain four increasing levels of strength: nominal, ordinal, interval and ratio.

The level of a measurement scale is nominal when the characteristics are only given in categories, i.e., white or black. It is ordinal

*The Data & Analysis Center for Software is a DoD Information Analysis Center operated by IIT Research Institute, Rome, New York under Contract No. F30602-83-C-0026, from the Air Force Systems Command, Rome Air Development Center, Griffiss Air Force Base, New York 13441.

[1] DACS Productivity Dataset contains summary information from over 400 software projects (productivity and error data, project duration, total effort, language and usage implementation technologies).

[2] NASA/SEL Dataset. The Software Engineering Laboratory (SEL) has been collecting software development data from NASA/SEL projects. The set contains over 45,000 records mainly from component status and run analysis reports. The remainder is comment information and change, resource summary and component summary reports.

0730-3157/83/0000/0017$01.00 © 1983 IEEE

when these categories can be ordered by a criterion, i.e., small, medium and large. It is an interval scale when it preserves the distance between two points even though the "zero" of the scale may be different for different scales measuring the same variable, i.e., the Kelvin and Fahrenheit temperature scales. Finally, it is a ratio scale if the zero is an absolute value, i.e., zero mass.

Statistical procedures are defined for different scales and can be correctly applied up to the scale for which the procedure is defined (see Figure 1). For example, contingency tables are defined for nominal scale variables, rank tests are defined for ordinal scale variables, and parametric tests that consider the calculation of distances (means, variances, residuals, etc.) are defined for variables given in at least an interval level of their measurement scale (SIEG56 and LEHM75). It is possible to lower the level of the measurement scale (i.e., take gross income data to income bracket data) and lose information in the process. The inverse process is not possible (i.e. take bracket information and specify income) without additional information.

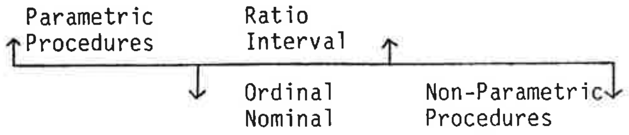


FIGURE 1: MEASUREMENT SCALE LEVELS AND STATISTICAL PROCEDURES

2.2 Analysis of the Productivity Data

An analysis task was defined for the DACS. It consisted in seeking any statistical relation between productivity and the usage of Modern Programming Practices (MPP's) in software development efforts. The data was examined and the literature surveyed for possible reference work. Several researchers ((NELS78), (TURN81), (BROO77), (CURT80a&b), (CURT79), (BASI81) and (WALS77), among others) had previously analyzed productivity[3] data pursuing different objectives. More recently, Dr. Barry Bohem has produced a comprehensive study (BOHE81) on software economics, deriving a model from a large set of software projects. These researchers had used parametric methods such as regression, correlation, and analysis of variance. It was therefore natural to start by looking at the data with similar procedures.

After our first analysis attempts, primarily using the regression model, several large violations of the regression model's assumptions surfaced, i.e.,

- non-normality of the residuals
- functional relation between mean and variance
- lack of fit for the established relation
- large and numerous outliers

In view of this situation, raw data was plotted and fitted to various known distributions, and the data was analyzed in detail. From these analyses several conclusions were obtained:

- variables were non-normal
- distributions were highly skewed
- large and numerous outliers were present
- a wide range of variability existed in the variables

This last point deserves further attention since it highlighted a very important fact. For two highly correlated variables (like size and effort), given the size, it was possible to find a project with twice the effort of another project of similar size. A similar situation occurred with size, given effort, to a minor extent. This situation prevailed even when the data was partitioned by language, dataset and usage of MPP's, i.e., into more homogeneous groups.

2.3 The Measurement Problem

Had the anomalies discussed above been the only ones found, the techniques of multivariate analysis, non-linear regression and variable transformation (among others) may have helped to cope with most of these situations. But the crucial problem was not any of the above-mentioned. Not only were there multiple variables in the analysis, some qualitative and some quantitative, not only were there problems of non-normality and heavy tails (outliers), not only were the variances of the residuals proportional to the response mean, but the measurement scale of the variables presented another even larger problem. As will be explained in the next section it was considered that the measurement scale of the variables involved in the present analysis attained at most an ordinal level.

2.3.1 The Problem of the the Measurement Scale Level

There are several reasons which have led us to believe that the analysis variables (i.e., project size, effort, productivity and the six MPP's) attain only an ordinal level of measurement scale.

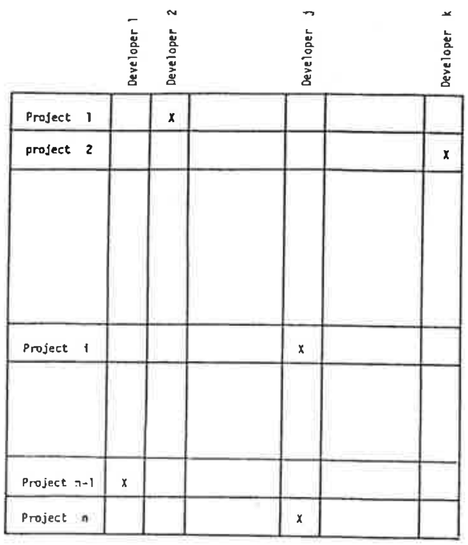
[3] Productivity data in the context of this paper covers software development data which include size (DSLOC) and effort (TMM). Size - The size of the software project in source lines. This count does not include unmodified reused code, test drivers or discarded code. This count does include reused code that was modified, internal program data and embedded comment lines. In some instances, estimates of the project size may have been rounded off. Effort - The effort spent on the development of the software project in man-months. In addition to coding activities, this figure includes the effort associated with the design, testing, and documentation of the project. In each instance this figure is recorded to the nearest man-month of effort.

These reasons are discussed in detail in (ROME82b) and can be summarized as:

i) characteristics of the activity

A software project is, in a way, a prototype by itself. It has been developed by a given organization in a given moment of the software development history (that defines the number and sophistication of software tools available) to solve a particular problem under very precise constraints. Hence, all the variables obtained (i.e., size, effort, operators, operands, etc.) may be a function of not only the inherent problem complexity which is being measured but also (and totally confounded with the former) the ability of the developer (Figure 2) plus some other undesired surrounding circumstances (Figure 3). These are the observed variables in software models.

Had we enough information about this problem (Figure 4) it could be possible to estimate the effects of these undesired (factors) surrounding circumstances. The fact that this is not possible due to lack of information lowers the level of the measurement scale of the observed variables.



	Developer 1	Developer 2		Developer j		Developer k
Project 1		X				
project 2						X
Project i				X		
Project n-1	X					
Project n				X		

X - represents the existence of an entry in that matrix cell.

FIGURE 4: EXAMPLE OF THE ORIGIN OF SOFTWARE DATA MEASUREMENT PROBLEMS

The current study analyzes data collected over an extended period of time. It includes a large number of different software projects and different software developers. The resulting data includes the effects of unaccounted-for factors in the model. This situation causes the lowering of the measurement scale of the "independent" variables down to an ordinal level. The parametric statistical methods previously employed are not designed for this measurement scale level.

ii) counting and round-off procedures

- Besides all of the factors stated in (i), variables like size and effort are rounded off to the higher man-month, the nearest thousand lines of code (LOC), etc., introducing more noise.

- For languages like Assembler (ASSY), conversion formulas are established to standardize the LOC value, which is contextually different for different environments.

- Not for all developers nor during the last 20 years has the concept of LOC been standard or had the same meaning for all people.

All of the above lead to the conclusion that the concept of distance is missing from the analyzed variables, i.e., that their measurement scale level is ordinal at most.

2.4 Proposed Solution

In view of the situation described in Section 2.2 and the considerations given in Section 2.3, the proposed solution was to assume that all analysis variables, i.e., project size (in DSLOC), project effort (in TMM), project productivity = size/effort, and the six MPP's, attain at most an ordinal level of their respective measurement scales. Since there is no way of increasing their measurement scale level without additional information (which was not available) we were forced to apply those statistical procedures which allow us to work with ordinal variables. These procedures are the non-parametric ones (Figure 1).

Hence, equivalent non-parametric procedures were substituted for the parametric procedures (see Table 1). The data was subjected to analysis by both methods (see Table 2). The risks and implications of correct model selection are graphically illustrated by Figures 5 and 6.

TABLE 1: EQUIVALENT PARAMETRIC/NON-PARAMETRIC STATISTICAL TESTS

PARAMETRIC TEST	NON-PARAMETRIC TEST	PURPOSE
Pearson	Kendall Tau	Tests Correlation
Simple Linear Regression	Non-Parametric Regression	Tests Trends
Student's t Test	Wilcoxon	Tests Location
F Test	Siegel-Tukey	Tests Dispersion

Finally, a simulation model was written and validated (Table 3) and used for comparing the efficiency of both procedures. The efficiency was evaluated on the ability of the procedure to filter out the noise introduced by the simulator. The nonparametric procedures came out ahead.

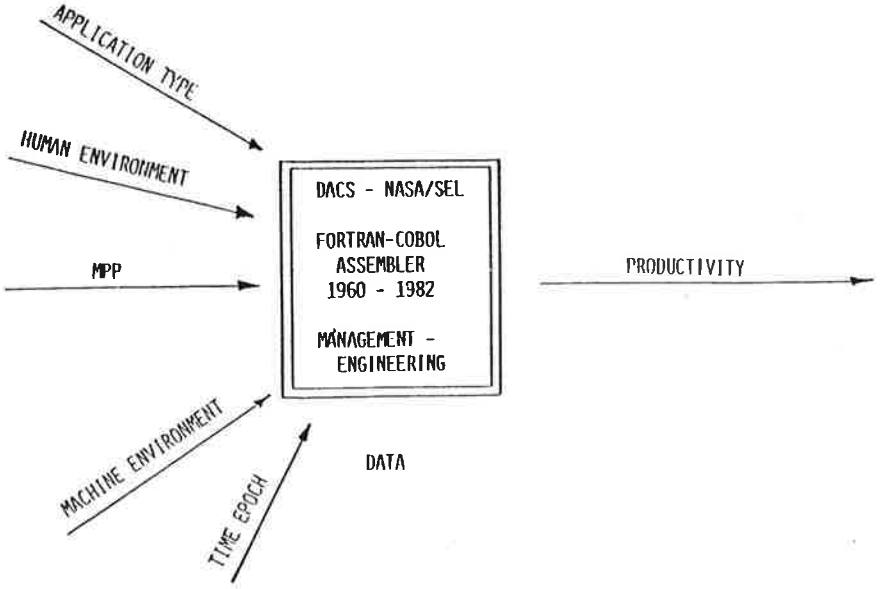


FIGURE 2: ANALYSIS OF AVAILABLE DATA MULTIPLE FACTORS

$$V_i = f(N_i(t), E_i(t), \ldots\ldots, ST_i(t))$$

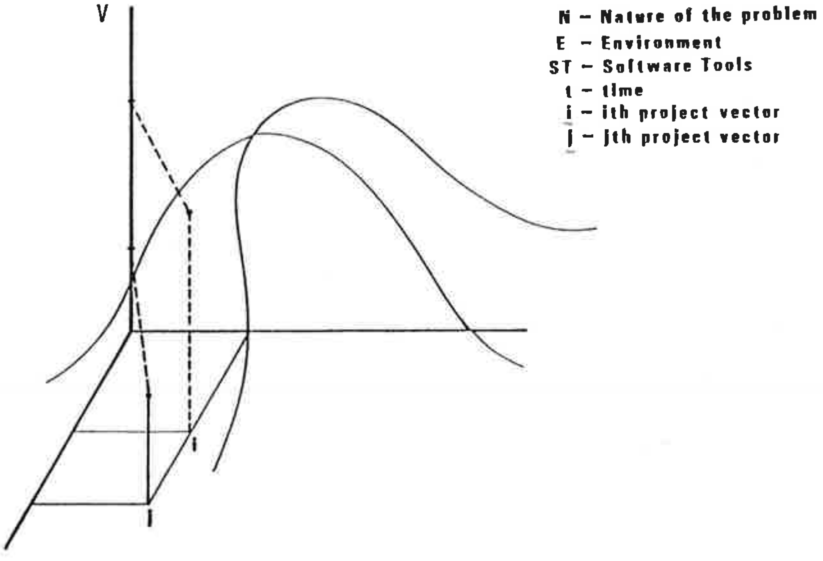


FIGURE 3: REPRESENTATION OF THE SOFTWARE MEASUREMENT PROBLEM

TABLE 2: COMPARISON OF PARAMETRIC AND NON-PARAMETRIC REGRESSION STATISTICS

	PARAMETRIC			NON-PARAMETRIC		
Group Examined	Pearson Correlation Coefficient*	Regression Slope	Regression Intercept	Kendall Correlation Coefficient	Regression Slope	Regression Intercept
DACS FORTRAN Projects (1) (2)	0.83 0.80	1.21 1.17	-3.164 -2.99	0.64 -	0.641 0.516	-0.735 -0.235
DACS FORTRAN Projects (MPP's Known)	0.71	0.699	-1.00	0.59	0.537	-0.29
DACS COBOL Projects	0.82	0.989	-2.49	0.52	0.954	-2.36
DACS COBOL Projects (MPP's Known)	0.85	0.987	-2.55	0.55	0.891	-2.05
NASA Projects	0.83	0.987	-2.66	0.74	1.040	-2.97
NASA FORTRAN Projects	0.89	1.006	-2.74	0.78	1.042	-2.98
NASA FORTRAN Projects (MPP's Known)	0.95	1.066	-3.07	0.92	1.050	-3.05

*Parametric and Non-Parametric Correlation Coefficients are not directly comparable because of different definitions and are included here for reference purposes only; both are significant at α = 0.01 for each case.

(1) including a 480K DSLOC project twice as large as any other in the dataset
(2) excluding this project

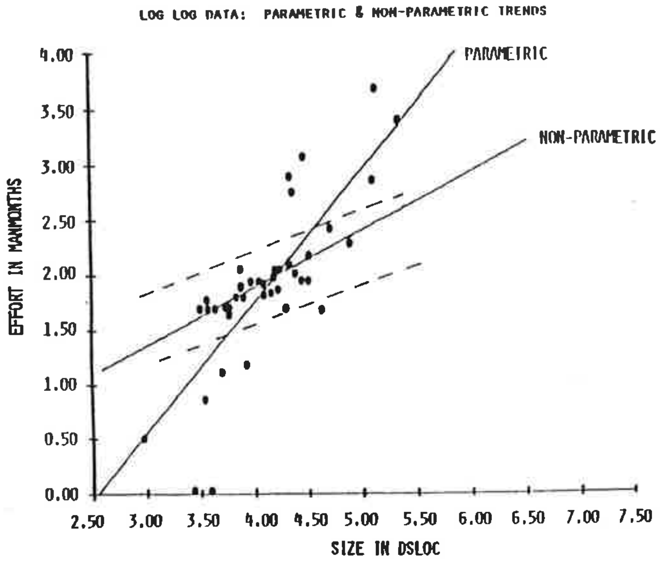


FIGURE 5: DACS FORTRAN PROJECTS - EFFORT VS SIZE

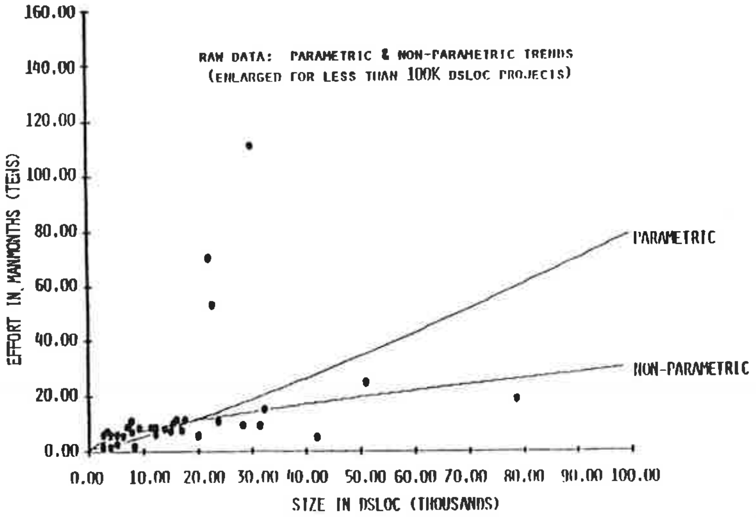


FIGURE 6: DACS FORTRAN PROJECTS - EFFORTS VS SIZE

TABLE 3: SIMULATION MODEL OBJECTIVES

1) Illustrate the feasibility of diagnosed problem
2) Compare the effectiveness of alternative procedures
3) Study and develop insight about the problem's underlying structure
4) Evaluate the quality of the data
5) Provide an alternative forecasting procedure

A detailed account of the analyses overviewed in this section can also be found in (ROME82b).

2.5 Statistical Consequences of Model Selection

Parametric models were employed previously by other researchers in the analysis of productivity data. Parametric models are not designed to deal with variables given in an ordinal level. On the other hand, the validity of the analysis results can no longer be guaranteed when the underlying[4] assumptions of a model are not met, hence, the importance of the selection of a good model in the presence of the data characteristics (Figure 7).

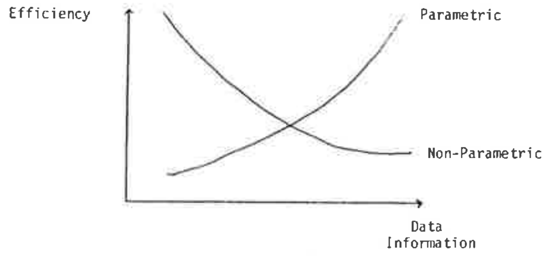


FIGURE 7: REPRESENTATION OF PARAMETRIC/ NON-PARAMETRIC TRADE-OFF

Grossly speaking, if the main differences between the parametric and non-parametric methods employed in (ROME82b) are classified into three broad categories as given in Table 4, the advantages of using a non-parametric approach in the present context are evident.

TABLE 4: SOME DIFFERENCES BETWEEN PARAMETRIC AND NON-PARAMETRIC ASSUMPTIONS AND METHODS

CATEGORY	PARAMETRIC	NON-PARAMETRIC
STATISTICAL DISTRIBUTION	NORMAL	CONTINUOUS
MEASUREMENT SCALE LEVEL	INTERVAL	ORDINAL
METHODOLOGICAL APPROACH	MINIMIZATION	SORTING

Figure 8 illustrates these advantages. Since the exact value of each variable is not known (for it is an ordinal value) the shaded region may be feasible for it.

Parametric regression states that:

i) Σe_i^2 is minimum
ii) e_i follow the normal distribution[4]

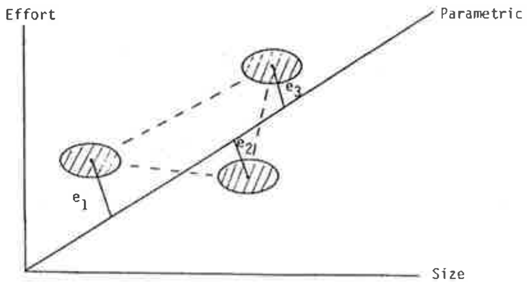


FIGURE 8: REPRESENTATION OF THE MODEL FIT

Figure 8 indicates how susceptible the parametric regression can be to any change in position of an e_i within its respective shaded region. It also indicates how much more resistant to these variations the nonparametric regression is. This criterion was used in the comparison of the efficiency of the two methods.

Table 4 can be interpreted as:

i) In the statistical distribution categories, it is no longer necessary to assume the residual's normality nor equality of variance, both highly suspect.

ii) In the measurement scale category, it is no longer necessary to assume an interval scale level, which is also largely suspect.

iii) Finally, since it is not assumed that (i) and (ii) are true, there is no point in "minimizing" the sum of squares e_i of a "distance."

In addition, the "non-parametric" estimator of the slope (SEN68) is obtained instead by sorting all possible slopes and selecting the median of this ordered set.

Finally, if for engineering purposes a confidence interval for a forecasted value is sought, a gross approximation can be obtained using the nonparametric probabilistic upper and lower bounds of the "nonparametric" slopes.

3.0 CONCLUSIONS

The fact that non-parametric statistics may be considered an appropriate approach for the

[4] The parametric regression assumptions are: the model is additive, the measurement scale is at least interval and the residuals are normally distributed with mean zero, equal variance and uncorrelated.

analysis of data, characteristic of developing activities like software engineering, is definitely not a shortcoming.

Non-parametrics is an area of very active statistical research, and great progress has been achieved in the past 10 years. There exist more than one non-parametric procedure for many univariate statistical analyses which are applicable under different circumstances. In addition, non-parametric methods for some multivariate analyses are also available.

Finally, non-parametrics only yield to their equivalent parametric procedures when the model's assumptions are met. This will eventually occur as the state-of-the-art in metrics and software activity in general advances. Until then, it may well be safer to play the non-parametrics card to provide the information needed as a basis for technology implementation decisions on a statistically justifiable basis.

4.0 ACKNOWLEDGEMENTS

Thanks are due to Rocco Iuorno (IITRI) for his encouragement and support and to Joseph Cavano (RADC) for reading the original version and for his helpful suggestions and criticisms.

REFERENCES

(BASI-81) Basili, V. "A Metamodel for Software Development Resource Expenditures," 5th International Conference on Software Engineering, 1981, pp 107-116.

(BOHE81) Bohem, B., Software Engineering Economics, Prentice-Hall, 1981.

(BROO-77) Motley, R.W., and W.D. Brooks, Statistical Prediction of Programming Errors, RADC-TR-77-175, May 1977.

(COMP-80) Basili, V. Tutorial on Models and Metrics for Software Management and Engineering, IEEE Computer Society, Oct. 1980.

(CURT-79) Curtis, B., S. Shepard, and P. Milliman, Experimental Evaluation of the Line Program Construction, TR-39-388100-6, December 1979.

(CURT-80a) Curtis, B., and P. Milliman, A Matched Project Evaluation of MPP's, Management Report on the Astros Plan, RADC-TR-80-6, Vol I/II 1980.

(CURT-80b) Curtis, B., S. Sheppard, and B. Kruesi, Evaluation of Software Life-Cycle from the Pave Paws Project, RADC-TR-80-28, 1980.

(LEHM-75) Lehman, E.L., Non-parametric Statistical Methods Based on Ranks, Holden Day, 1975.

(NELS-78) Nelson, R., "Software Data Collection and Analysis at RADC," Draft report, RADC, 1978.

(ROME-82a) Romeu, J. L., and C. Turner, "An Investigation of the Effects of Technology on Development Efforts," 5th Minnowbrook Workshop on Software Performance Evaluation, July 1982.

(ROME-82b) Romeu, J. L., "An Investigation of Parametric vs Non-parametric Techniques for the Analysis of Software Engineering Data," Technical report (in process) Data & Analysis Center for Software.

(SEN68) Sen, P.K., "Estimates of Regression Coefficients Based on Kendall's Tau" JASA. Vol. 63, pp. 1379 - 1389.

(SIE6-56) Siegel, Sidney, Non-parametric Statistics for the Behavioral Sciences, McGraw-Hill, 1956.

(TURN-81) Caron, G., and C. Turner, "A Comparison of RADC-NASA/SEL Software Development Data," DACS Technical Monograph, May 1981.

(WALS-77) Walston, C.E., and C.P. Felix, "A Method of Programming Measurement and Estimation," IBM Systems Journal Vol. 16, No. 1, 1977, pp. 54-65.