AN EMPIRICAL ASSESSMENT OF PROJECT DURATION MODELS IN SOFTWARE ENGINEERING

Serge Oligny[1], Pierre Bourque[2] , Alain Abran[2]

ABSTRACT

Many parametric models based on estimates of project effort have been proposed in the literature to predict the duration of software development projects. Among these, COCOMO has received wide attention. A comparison of the duration estimates obtained from this model with those from an empirical model derived from a set of historical data maintained by the International Software Benchmarking Standards Group (ISBSG) is presented in this paper. It is shown that the COCOMO duration estimates are "optimistic" when compared to the empirical model estimates. Using quantitative evaluation criteria, this paper also shows that the goodness of the COCOMO duration models is very close to the goodness of the empirical model in spite of fact that the data used to derive the COCOMO duration models are roughly 20 years old.

KEYWORDS

Software Engineering, Project Duration Models, ISBSG, COCOMO

1. INTRODUCTION AND CONTEXT

Complete and stable product requirements, high quality, low costs and a short time to market are probably the four most prized characteristics of the ideal software development project. Many managers will tell you that, given adequate control over these four characteristics, time to market is the hardest one to pin down. It is the object of much attention from the project users and customers, and a variation of any of the other three characteristics has a determining influence on it.

Although many additional factors might have an influence on the value of each of these four characteristics, it is generally recognized that requirements determine product size. Product size in turn determines project effort, and, finally, project effort determines project duration. The viewpoint adopted in this paper centers on the planning of software development project duration with the assumption that a valid effort estimate is available.

Many parametric models based on project effort have been proposed in the literature to predict the duration of software development projects. Among these COCOMO [1] has received wide attention in the literature, and we will refer to it as the reference model. This paper presents an analysis and a comparison of the duration estimates obtained from this reference model with a duration model developed using data (250 projects) collected by the International Software Benchmarking Standards Group (ISBSG) [2], as of June 1996.

It is shown that most of the estimates obtained with the reference model are "optimistic" when compared to those obtained with the duration model from the ISBSG-1996 dataset. Using quantitative evaluation criteria, this paper also shows that the goodness of the COCOMO duration models is very close to the goodness of the empirical model in spite of fact that the data used to derive the COCOMO duration models are roughly 20 years old.

Section 2 of this paper presents the selected data sample. Section 3 presents the derived duration model and section 4 compares the results of this

[1] Serge Oligny is Corporate Software Development Manager, Alliance Forest Products, 1000 de la Gauchetière Ouest, Suite 2820, Montréal (Québec) Canada, H3B 4W5, Phone:1 514-954-2157, Fax: 1 514-954-2180, email: Serge.Oligny@alliance-forest.com

[2] Pierre Bourque and Alain Abran are, respectively, Assistant Director and Director of the Software Engineering Management Research Laboratory, Département d'informatique, Université du Québec à Montréal, C.P. 8888, succ. Centre-Ville, Montréal (Québec) Canada H3C 3P8, Phone: 1 514-987-3000 ext: (0315), Fax: 1 514- 987-8477, email: bourque.pierre@uqam.ca; abran.alain@uqam.ca

model with those of COCOMO. A brief conclusion is presented and further opportunities for research are identified in section 5.

2. ANALYSIS OF THE DATA SAMPLE

Credible empirical analysis must be based on a large sample of observations. Gaining access to a collection of observations of adequate size is, more often than not, problematic. This statement is also true for a large number of practitioners working in commercial environments.

These practitioners are usually placed in the position of either: a) collecting their own historical sample the utility of which will not become apparent in the short term, or b) being ready to pay a hefty entry price to access specialized expertise based on the commercialization of proprietary databases.

An alternative has appeared recently with the availability of an open repository from the International Software Benchmarking Standards Group[3] (ISBSG). This group focuses on collecting, validating and publishing a repository of historical software development project productivity data. In June 1996, ISBSG published the third release of their repository which contained historical data on 323 software development projects completed between 1989 and 1996.

These projects were conducted in 13 countries throughout Asia-Pacific (65%), North America (21%) and Europe (14%). The majority of projects were intended for use in organizations of type: public administration (20%), finance, property and business services (18%), banking (13%) and manufacturing (11%). Project development types are broken down into new development (66%), enhancements (30%) and redevelopment (4%). The three main types of applications are transaction/production (40%), MIS (28%) and office information (13%). Almost three-quarters (74%) of the projects were developed in-house for intended use by internal business units. One third of the projects involve a client/server architecture. Almost two-thirds (65%) involve a mainframe platform. 3GL development languages are used in 38% of projects, 4GL in 43% of projects.

The empirical duration model presented in this paper is based on this data set.

[3] See www.bs.monash.edu.au/asmavic/isbsg.htm

2.1 Basic criteria

Among the 323 projects of the ISBSG-1996 repository, projects showing the following characteristics were selected:

- No reasonable doubt as to data validity; that is the ISBSG has not flagged this project as having uncertain data and has retained it for its own analyses;
- Known effort
- Known duration;
- Effort greater than or equal to 400 man-hours.

The first three criteria are easy to understand. The fourth criterion was chosen on the basis that endeavors involving fewer than 400 man-hours are often, in the experience of the authors, considered too small to be the object of a formal project structure in many organizations. 250 projects satisfied all of these criteria. Basic descriptive statistics are shown in

.

	Duration (D) ***in calendar-months***[4]	**Effort (E)** ***in man-hours***
Number of observations	250	250
Minimum value	1,0	400
Maximum value	78,0	106480
Mean value	11,6	6925
Standard deviation	9,4	13203
Median	9,0	2535

Table 1 Descriptive Statistics of Sample

Project duration ranges from one calendar-month to 78 calendar-months (6,5 calendar-years). Assuming that an average man-month is equivalent to 140 man-hours[5] and 12 such man-months are equivalent to a man-year, project effort ranges from 3 man-months to 761 man-months (63 man-years). A scatter plot of the data is shown in Figure 1.

It is held that both ranges cover the majority of software development projects conducted by corporate IT business units today.

[4] See [2] for a detailed discussion on the recording levels and the recording methods for Effort.

[5] 140 man-hours per man-month can be considered representative of the Canadian software industry

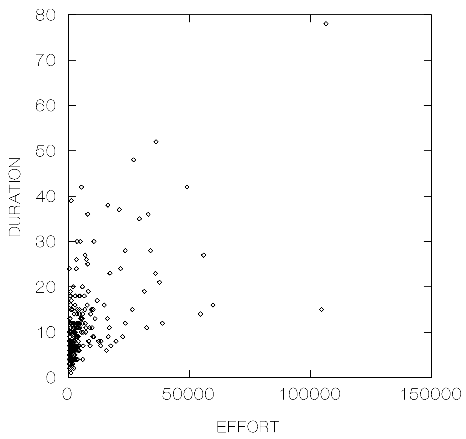

Figure 1 - Scatter plot of project effort (E) in man-hours vs. duration (D) in calendar-months (n=250)

2.2 Analyzing the distribution of sample data

Figure 1 indicates that the regression function of Duration vs. Effort is non-linear and that a greater extent of scatter is observable for projects requiring much effort. Distributions of effort and duration data in the 250 projects of the sample are shown in Figure 2 and Figure 3.

As can be seen visually from the superimposed normal curves in Figure 2 and Figure 3, both distributions are : a) skewed to the left, and b) "cut-off" on the left.

Distribution skewness simply reflects the fact that the sample contains more small projects than large ones. It is our experience that smaller projects are undertaken more frequently than larger ones in most organizations, and the ISBSG repository reflects this.

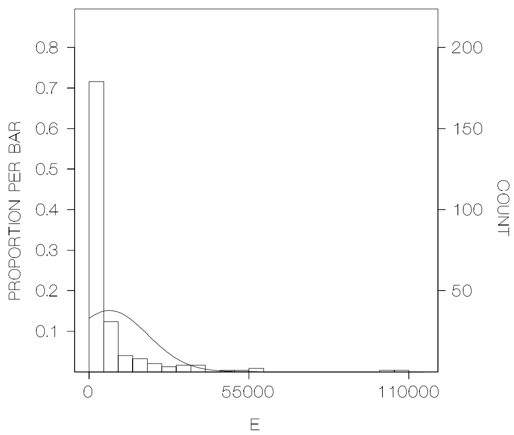

Figure 2 - Distribution of Effort (man-hours) (n=250)

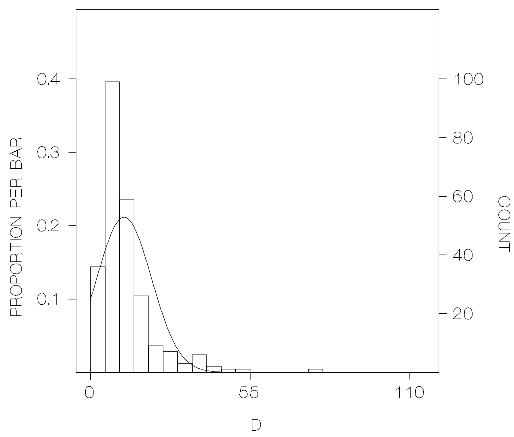

Figure 3 - Distribution of DURATION (calendar-months) (n=250)

The "cut-off" on the left of both distributions is imposed by the very nature of the data. An observation with effort showing a value that is less than or equal to zero does not make sense. Furthermore, all projects showing an effort of fewer than 400 man-hours have been eliminated. The same argument applies to duration, since a value less than or equal to zero does not make any practical sense either.

2.3 Sample retained for modeling

Taken as is, such data distributions offer weak support for linear regression. It was thus deemed appropriate to work on a mathematical transformation of the data. A LOG transform was therefore applied. Figure 4 and Figure 5 show the resulting distributions.

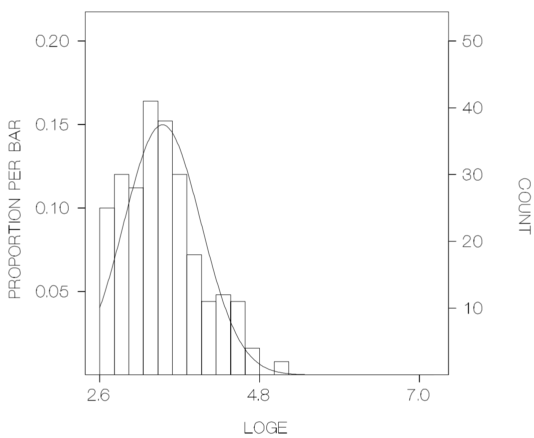

Figure 4 - Distribution of LOG(Effort)

The distribution was tested for normality using a skewness statistic ($\sqrt{b_1}$) and a kurtosis statistic (b_2). A further "omnibus" test (K^2), able to detect deviations from normality due either to skewness or to kurtosis was calculated. As suggested by D'Agostino *et al.* in [3], these tests are deemed to be powerful and informative. Results are presented in Table 2 (before LOG transform) and in Table 3 (after LOG transform).

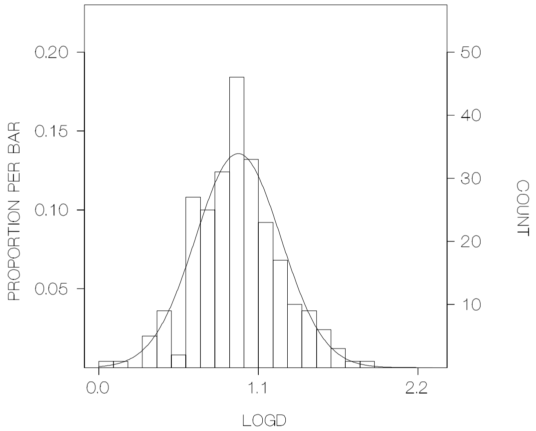

Figure 5 - Distribution of LOG(Duration)

Table 3 shows that the LOG transform was successful in the sense that the LOG(Duration) data follows a normal distribution. Normality test results confirm this at the 0,05 confidence level.

Table 3 also shows that the LOG transform on the effort data has increased the normality of the distribution. As shown in Figure 6, the LOG transformation has led to a linear regression function and has reduced the extent of scatter for high levels of Effort that was observable in Figure 1. The LOG-transformed values will therefore be used to derive an empirical model linking project effort to project duration.

Variable	Stat.	Value	Sign. (α<= 0,05)
Effort	$\sqrt{b_1}$	4,486	YES[6]
	b_2	28,362	YES
	K^2	161,89	YES
Duration	$\sqrt{b_1}$	2,800	YES
	b_2	14,591	YES
	K^2	115,22	YES

Table 2 - Results of normality tests BEFORE LOG transform

[6] Hypothesis of normality is rejected at the 0,05 level.

Variables	Stat.	Value	Sign. (α <= 0,05)
Log(Effort)	$\sqrt{b_1}$	3,256	YES[6]
	b_2	-0,407	NO[7]
	K^2	10,77	YES
LOG(Duration)	$\sqrt{b_1}$	0,734	NO
	b_2	1,463	NO
	K^2	2,678	NO

Table 3 Results of normality tests AFTER LOG transform

3. DERIVING AN EMPIRICAL MODEL

Deriving an empirical model from the LOG transform sample was performed in two steps: first, a correlation analysis was conducted; then, a regression analysis supplied the value of the coefficients.

3.1 Correlation analysis

Figure 6 shows a scatter plot of the LOG-transformed variables. The Pearson correlation coefficient between LOG(effort) and LOG(duration) is 0,592. Given the size of the sample (n=250) it is significant at the 0,05 level.

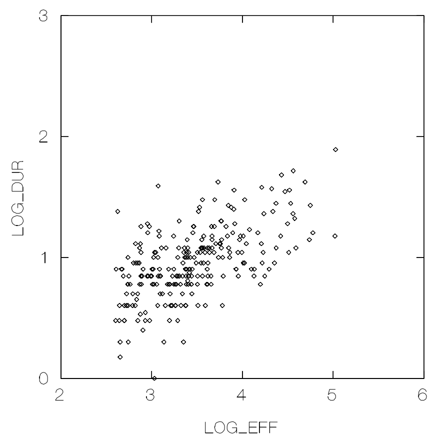

Figure 6 - Scatter plot of LOG-transformed variables, effort and duration (r = 0,592)

3.2 Regression analysis

Four hypotheses must be confirmed in order to accept the results of a regression analysis:

- linear relation is judged adequate ;
- residuals are normally distributed ;

[7] Hypothesis of normality is NOT rejected at the 0,05 level.

- residuals are independent from the independent variable ;
- variance of residuals is constant.

Bearing this in mind, a linear regression has been performed on the LOG transformed values of effort and duration using LOG(effort) as the independent variable. Two projects were removed from the sample because they showed a high leverage on the regression results. Five other projects were removed because they showed large studentized residuals (outliers). Table 3 summarizes the reasons for removing these seven projects.

Selected variables for these seven projects are presented in Table 4 below.

Analysis of these variables reveals that project 30133 shows the largest workload of the sample, and has the longest schedule of the sample. Project 20021 contains the second largest value for effort. Both projects delivered fairly large applications (between 4000 and 5000 function points). These two projects are isolated at the higher end of the sample.

ISBSG Project ID	Reason for rejection	Value
20021	High leverage	0,038[8]
30133	High leverage	0,039[8]
29051	High studentized residual	2,985[9]
30074	High studentized residual	-2,661[9]
25024	High studentized residual	-3,541[9]
29008	High studentized residual	3,266[9]
30115	High studentized residual	2,660[9]

Table 4 - Projects rejected from regression sample with reason for rejection

ISBSG Project ID	Effort	Duration	Size (Function Points)	Max. team size
30133	106480	78	4913	24
20021	104690	15	4181	unknown
29051	423	24	496	unknown
30074	2244	2	276	unknown
25024	1073	1	64	3
29008	1177	39	378	2
30115	5400	42	826	3

Table 5 - Selected characteristics of rejected projects

Projects 30074 and 25024 both show the characteristics of “crash projects”, since a relatively large effort was expended in a short time, leading to the conclusion that a large number of individuals were rushed into delivering the results. At an average rate of 140 man-hours per month, project 30074 required 8 resources to deliver a relatively small project (278 function points). Considering an average month of 20 working days, project 25024 extracted from its 3 resources an average of 18 hours of work per working day, or 12 hours per day, if working during weekends is considered! All that effort to deliver a meager 64 function points.

At the opposite end of the spectrum, projects 30115, 29051 and 29008 are considered “part-time” projects since the average spread of the effort over the duration shows the consumption of a fraction the availability of dedicated resources. Project 30115 shows the consumption of only 31% of the availability of the 3 resources, project 29008 shows a even lower 11% consumption of the 2 available resources. Project 29051 shows an average consumption of 0,125 resource over its duration (equivalent to 1/8 of the effort of a full-time resource).

In all seven cases, it is thus held, based on our knowledge and experience, that the prevalent context is not representative of the typical MIS project. Therefore, these projects were not included in the final regression analysis.

---

[8] h_i : leverage coefficient

[9] e^*_i :studentized coefficient

SELECTED RESULTS	VALUE
Sample size (n):	243
R^2:	0,381
F(1,241):	148,899
Prob > F :	0,0001
Log(E) coefficient:	0,328
Standard error of Log(E):	0,027
Constant:	-0,179

Table 6 - Selected results from the regression

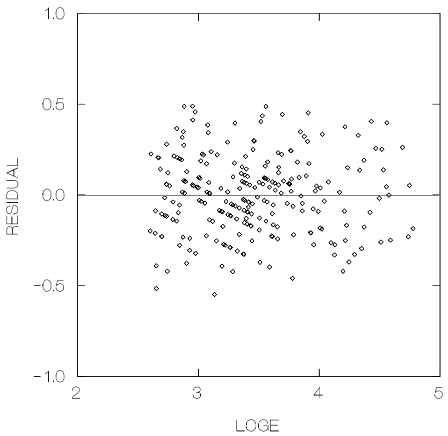

Figure 7 - Residuals against LOG(Effort) (independent variable)

Regression results are shown in Table 6. Analysis of residuals against both LOG(effort) and LOG(estimated duration) are shown in Figure 7 and Figure 8.

Figure 7 shows that the residuals are randomly distributed over the range of the independent variable (LOG(effort)). Figure 8 shows that the variance of the residual is also constant over the range of the dependent variable (LOG(Duration)).

Judging that a linear model is adequate and since the other 3 hypotheses of linear regression are satisfied, it is held that the resulting model is acceptable.

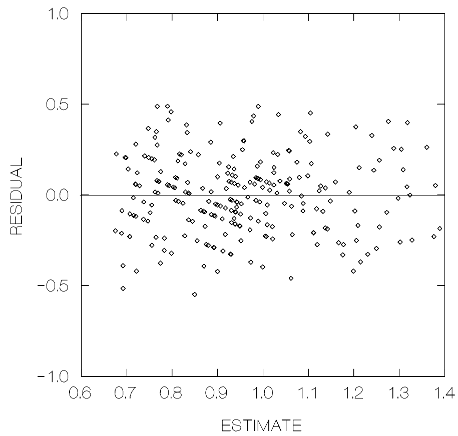

Figure 8 - Residuals against LOG(estimated duration) (dependent variable)

The empirical model linking project effort and duration can thus be characterized by the following equation:

$$LOG(D) = (0,328*LOG(E_{man\text{-}hours})) - 0,179$$

or, in the traditional form:

$$D = 0,662 * E^{0,328} \quad \text{(E in man-hours)}$$

4. COMPARING WITH COCOMO

COCOMO duration equations[10] [1] depend on the type of development. These equations are formulated for effort expressed in man-months. Using a constant of 152[11] man-hours per man-month, to be able to compare COCOMO equations with the empirical model, these same equations can be reformulated as:

Embedded	$D = 0,501 * E^{0,32}$	(E in man-hours)
Semi-detached	$D = 0,431 * E^{0,35}$	(E in man-hours)
Organic	$D = 0,371 * E^{0,38}$	(E in man-hours)

Figure 9 depicts how the empirical model compares with the COCOMO equations. Furthermore, using the criteria suggested by Conte *et al.* in [4], an evaluation of each LOG transformed model is shown in Table 7.

Figure 9 reveals that the COCOMO duration estimates tend to be lower than the duration estimates produced by the empirical model.

[10] See Section 5.1 Basic Definitions and Assumptions of [1] for a detailed discussion on what is included in Effort and Duration.

[11] See Section 5.1 Basic Definitions and Assumptions of [1]

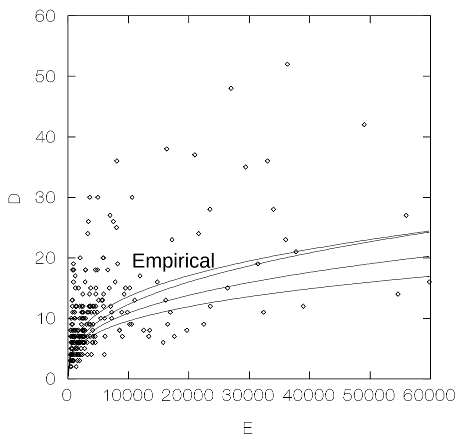

Figure 9 - Graphical comparison of empirical model (upper curve) with COCOMO duration models (lower three curves)

Criteria	Emp	C_{emb}	C_{sd}	C_{org}
R^2	0,381	0,381	0,381	0,381
Rel. Pos.[12]	1	1	1	1
Avg. RE	-0,063	0,108	0,067	0,026
Rel. Pos.	2	4	3	1
Avg. MRE	0,211	0,219	0,209	0,204
Rel. Pos.	3	4	2	1
Pred (0,25)	75 %	64 %	69 %	70 %
Rel. Pos.	1	4	3	2
RMS	0,214	0,262	0,242	0,228
Rel. Pos.	1	4	3	2
RMS bar	0,224	0,274	0,253	0,239
Rel. Pos.	1	4	3	2

Table 7 - Evaluation of the LOG transformed empirical model (Emp) and the LOG transformed COCOMO (C_{xxx}) models using CONTE *et al.* criteria[13]

Conte *et al.* propose that only models with an average MRE less than or equal to 0,25 while simultaneously displaying a Pred(0,25) greater or equal to 75% can be considered as having an acceptable performance. Table 7 shows that only the LOG transformed empirical model simultaneously meets these two conditions while the LOG transformed COCOMO models come relatively close. In other words, all four models produce estimates with an average relative error of less than or equal to 25%. However, only the LOG transformed empirical model also produces 75% of its estimates with a mean relative error less than or equal to 25%.

However, it should be noted:

- that all LOG transformed models do not explain more than 38% (Table 7, R^2) of the variance of the dependent variable. The largest proportion of the variance in duration is therefore not accounted for by the project effort. A more accurate estimation model for duration must at least include another significant variable ;
- that the values of the evaluation criteria are very close for all four models ;
- that the values for the evaluation criteria (except R^2) for the models (without the LOG transform) are substantially different than those presented in Table 7. The goodness of the LOG transform models are much better than the goodness of the not transformed models due to the fact that the LOG transform is not a linear nor a constant transformation.

5. CONCLUSION AND FURTHER RESEARCH

Roughly 20 years separate the projects used to develop COCOMO from the ones used to derive the empirical duration model presented in this paper. Based upon the ISBSG-1996 data set, this paper shows that:

- the COCOMO duration equations still compare favorably with the empirical model developed with the ISBSG-1996 data set ;
- COCOMO duration estimates tend to be lower than the estimates produced by the duration model developed with the ISBSG-1996 data set ;
- none of the models explain more than the 38% of the variance of the project duration data.

In the perspective of these conclusions, many axes of research should be pursued. A comparison with other known “effort-duration” models should be completed. Second, the size of the sample permits the subdivision of the ISBSG data set into more homogeneous data sets using characteristics like project type (new developments or major enhancements), or business area (banking, manufacturing, telecommunications, etc.). Better context-sensitive project duration models, for example explaining more of the variance, could possibly be derived from these subsets of data. The usage of non-linear models and the incorporation of other variables in the modeling process should also be

[12] Rel. Pos.: Relative Position

[13] R^2: coefficient of multiple determination, RE: relative error, MRE: mean relative error, Pred: % of prediction at 25% level or better, RMS: root mean square error, RMS bar: relative RMS.

explored. A comparison with COCOMO 2.0 should also be conducted [5].

Using the software productivity data that was used to develop COCOMO roughly 20 years ago and the more recent ISBSG data, an another worthwhile topic to pursue would be to conduct a comparative analysis to determine if the software engineering field has developed and implemented methods and techniques which enable more work to be done in parallel. If so, then the analysis should indicate than more project effort is expended in a same project duration

Finally, a comparison with "effort-duration" equations from other fields, civil engineering for instance, might provide some clues as to the relative degree of maturity of software engineering as a discipline.

ACKNOWLEDGMENTS

This research was carried out at the Software Engineering Management Research Laboratory, at the Université du Québec à Montréal. The laboratory is supported through a partnership with Bell Canada. Additional funding is provided by the Natural Sciences and Engineering Research Council of Canada. The authors would like to thank Bertrand Fournier for his statistical counseling. The opinions expressed in this paper are solely those of the authors.

REFERENCES

[1] B. W. Boehm, *Software Engineering Economics*: Prentice Hall, 1981.

[2] ISBSG, " Worlwide Software Development - The Benchmark," International Software Benchmarking Standards Group, Victoria, Australia Release 3, July 1996.

[3] R. B. D'Agostino, A. Belanger, and R. B. J. D'Agostino, "A suggestion for using powerful and informative tests of normality," *The American Statistician*, vol. 44, 1990.

[4] S. D. Conte, H. E. Dunsmore, and V. Y. Shen, *Sofware Engineering Metrics and Models*. Menlo Park, CA: Benjamin Cummings, 1986.

[5] B. Boehm, B. Clark, E. Horowitz, C. Westland, M. R., and S. R., "Cost models for future software life cycle processes: COCOMO 2.0," *Annals of Software Engineering*, vol. 1, pp. 57-94, 1995.