World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:4, No:2, 2010

Comparison of Imputation Techniques for Efficient Prediction of Software Fault Proneness in Classes

Geeta Sikka, Arvinder Kaur Takkar, and Moin Uddin

Open Science Index, Computer and Information Engineering Vol:4, No:2, 2010 publications.waset.org/9138/pdf

***Abstract*—**Missing data is a persistent problem in almost all areas of empirical research. The missing data must be treated very carefully, as data plays a fundamental role in every analysis. Improper treatment can distort the analysis or generate biased results. In this paper, we compare and contrast various imputation techniques on missing data sets and make an empirical evaluation of these methods so as to construct quality software models. Our empirical study is based on NASA's two public dataset. KC4 and KC1. The actual data sets of 125 cases and 2107 cases respectively, without any missing values were considered. The data set is used to create Missing at Random (MAR) data Listwise Deletion(LD), Mean Substitution(MS), Interpolation, Regression with an error term and Expectation-Maximization (EM) approaches were used to compare the effects of the various techniques.

***Keywords*—**Missing data, Imputation, Missing Data Techniques.

I. INTRODUCTION

ACCURATE prediction of Software Fault proneness in a class is a challenging task. Researchers using data sets for prediction of software fault proneness are often confronted with incomplete data sets. The problem of missingness has been an area of concern in not only software quality but in all areas of research including education, medicine, nursing [6], economics [7] and marketing [8]. Data Analysts and researchers mostly have to ignore the entire record values even if one of the attribute values is missing. Simply deleting the cases with missing values, may lead to extensive biases in the analysis. They may carry some useful information. This incompleteness in datasets is an unrelenting problem for them. Proper decision making or knowledge discovery in large data sets cannot be made because of this problem. The cause of incompleteness may occur due to none reporting, by chance or may be intentional. There are various Imputation methods for handling missing data. Imputing means to fill in missing data with a value, but before filling in the value or obtaining the estimates, the types of missingness should be known.

Geeta Sikka is Asstt. Professor in the Department of Computer Science and Engineering at Dr. B R Ambedkar National Institute of Technology, Jalandhar, 14401 1, Punjab, India (phone: 0181-2690232; 9888582299; e-mail: ikkag@gmail.com).

Arvinder Kaur is Associate Professor in the School of Information Technology at Guru Gobind Singh Indrapratha University, Delhi, India (e-mail: arvinderkaurtakkar@yahoo.com).

Moin Uddin is Director at Dr. B R Ambedkar National Institute of Technology, Jalandhar, 144011, Punjab, India (e-mail: director@nitj.ac.in).

Missing value mechanism was introduced by Little and Rubin[1] and Schafer[3].According to them, if the missingness depends on both observed and missing values, then it is not missing at random(NMAR), if the missingness is dependent on only observed values and not on missing values, then it is missing at random(MAR). Missingness may neither depend on observed nor on missing values, in that case the values are missing completely at random (MCAR). NMAR values cannot be ignored whereas MCAR and MAR values can be ignored. In MCAR and MAR the data are recoverable, whereas in NMAR the missing data is not recoverable.

The missing data problem has been addressed in Statistical Literature [4,5]. In this paper, the techniques used for handling missing data are Casewise Deletion method(CWD) and Mean Substitution(MS).It focuses on ignoring the cases having missing values or replacing all missing values by the mean value of the variable.

Strike et al. [10] performed a simulation study on three techniques: Listwise Deletion (LD), Mean or Mode Single Imputation (MMSI) and eight Hot Deck Single Imputation techniques (HDSI) in context of Software Cost Modelling.

A comparative study on Missing Data Techniques has been carried out by Myrtveit [9] in the context of Software Cost Estimation. The study was carried out on 176 projects. The techniques used were Listwise Deletion (LD), Mean or Mode Single Imputation (MMSI), Similar Response Pattern Imputation (SRPI) and Full Information Maximum Likelihood (FIML).

Cartwright et al. [11] showed the performance of k–Nearest Neighbour Single Imputation (kNNSI) and Sample Mean Imputation (SMI) on two industrial sets and inferred that kNNSI yielded better results than SMI.

A Multiple Imputation (MI) technique has been emphasised by Rubin [2]. In this approach the missing values are imputed conditional on the non missing values. The details of MI are given in Rubin [2] and Schafer [3].

Carol et al. [6] also made Comparison of Imputation techniques for Handling Missing Data. They compared and contrasted the limitations of five Missing Data Techniques, including Regression with error term and Expectation-Maximization.

Twala et al. [12] makes a comparison of seven MDT's using eight datasets and inferred that Listwise Deletion is the least effective while Multiple Imputation is the most effective of all techniques. He suggested an algorithm by a combination

Open Science Index, Computer and Information Engineering Vol:4, No:2, 2010 publications.waset.org/9138/pdf

of MDT's which lead to remarkable prediction for missing values.

Twala [13] also investigated the robustness and accuracy of seven missing data techniques. The seven techniques were compared by simulating different proportions, patterns and different mechanisms of missing data using 21 data sets. Besides the strengths and weaknesses of various techniques, LD was considered the least efficient and Multiple Imputation that uses EM was proved to be the most effective.

The purpose of our study is to predict the missing values on fault proneness in classes. As already discussed there are numerous techniques used to handle MCAR and MAR missing data. Some well known methods that are used in this paper for the prediction of Software Fault Proneness Models are: Listwise Deletion, Mean substitution, Interpolation, Expectation Maxmization and Regression.

List Wise Deletion:

The Simplest approach is the Listwise Deletion which means to completely ignore the tuples with missing data and to run the analysis on what remains. This leads to a decrease in the sample size which is available for analysis. This method is still used by engineering researchers, because of its easiness and simplicity.

Mean Substitution:

In mean substitution method, the attribute mean is used to fill in the missing values. In this approach, the mean of the given attribute, replaces the missing values. This is a very simple and efficient method but gives biased estimate. Use the attribute mean or median for all samples belonging to the same class. With mean substitution if we are missing a person's height, weight or income we substitute the average .Thus, the overall mean with or without replacing the missing data, will be same. Only the sample size has increased and the standard error is reduced. According to Rubin [1] mean imputation, decreases the variability in the dataset, because mean is used as a substitute for all the missing values.

Interpolation:

This method is also used in our study. It is a method of constructing new points within the range of known data points. We are using Interpolation for treating missing data in datasets.

Regression Estimation:

By using Regression, the missing values are predicted on the basis of other variables. The missing values that are calculated depend conditionally on other information that is available. The variable with missing data is treated as dependent variable; where as the other variables are treated as Independent variables. The regression equation is generated that is used to predict the missing values. In the regression equation,

$$Y = bX + a \tag{1}$$

first thing is to estimate the regression with whatever data is available and then the X values are used to find the missing Y values. In the regression equation b is the slope of the line and a is the Y-intercept. Slope is given by the formula:

$$b = \sum\left[(x_i - \bar{x})(y_i - \bar{y})\right] / \sum (x_i - \bar{x})^2 \tag{2}$$

and the intercept is

$$a = \bar{y} - b\bar{x} \tag{3}$$

In the mean substitution method, if the income of a person is missing, we substitute it with the average income, while in regression substitution of the average income of the person will be from the same profession or from the same age group. This method is an improvement over mean substitution but the problem of error variance still exists. By substituting a value that is absolutely predictable from other variables, we have not really added more information but we have increased the sample size and reduced the standard error. In order to reduce the problem, a bit of random error is added to each substitution.

Expectation Maximization:

The best known method that is used for missing value imputation is the Expectation–Maximization. This is generally called as the EM algorithm. This is a very powerful technique. According to this technique, if the missing values for the tuples are known, then the model parameters can be estimated. Correspondingly, if the parameters of the data model are known, the missing values can be obtained. First thing is to estimate the regression with whatever data is available and then the X values are used to find the missing Y values. The missing values are then filled in the data and regression coefficients are recalculated. EM is based on iterating the process of regression imputation. It does two things; firstly estimating the missing data based on the parameters and later re-estimate the parameters based on the missing data that were filled. Once the missing data is filled, then the regression coefficients are recalculated on the entire data sets. The EM algorithm adds a bit of errors to the variances it estimates, and uses estimates to impute data, and this continues until the solution stabilizes. Maximum likelihood estimates of the parameters are thus obtained, and can be used to make the final maximum likelihood estimates of the regression coefficients.

In this study, the binary dependent variable is fault proneness. The goal is to empirically investigate the effect of missing data on the predictions. The simulation study performs analysis with the various missing data techniques (MDT's) on two public data sets. The actual NASA KC4 dataset of 125 observations and NASA KC1 dataset with 2107 observations having no missing values were used to create data sets with 10-30% missing data at random(MAR). The missing values for the binary dependent variable fault proneness were predicted by using various missing data techniques (MDT's). The results of the various techniques were compared. After investigating the results, discussions and improvisation for future research are laid down.

Open Science Index, Computer and Information Engineering Vol:4, No:2, 2010 publications.waset.org/9138/pdf

TABLE I
DATASETS USED FOR EXPERIMENT

Dataset	Observations	Attributes	
NASA KC4	125	9 Numeric	Software Metrics and Fault proneness
NASA KC1	2107	17 Numeric	

II. RESEARCH METHODOLOGY

We compare the various imputation algorithms over different percentages of missing data values. 10-30% data was randomly removed from both the data sets and values were imputed with MDT's. The main objective is to find the accuracy and significance of the MDT used in the study.

Dependent and Independent Variables: The binary dependent variable in our study is fault proneness. Fault proneness is the probability of fault detection in a class. The independent variables in the two data sets are the various Software metrics (method-level and class-level). The goal of our study is to predict the value of dependent variable using the various MDT's.

Empirical Data Collection

This study makes use of two public domain data sets KC4 and KC1 from the NASA Metrics Data Set Repository.

Data Set 1: The data in KC4 consists of 25 KLOC of Perl source code. This system consists of 125 classes and provides method–level static metrics. Table II gives the list of metrics used in the data set.

TABLE II
THE NINE INDEPENDENT VARIABLES USED IN KC4 DATASET

No	VARIABLE
1	BRANCH_ COUNT
2	CALL_ PAIRS
3	CYCLOMATIC_ COMPLEXITY
4	DESIGN_ COMPLEXITY
5	DESIGN_ DENSITY
6	EDGE_ COUNT
7	MAINTENANCE_SEVERITY
8	NODE_COUNT
9	LOC_COUNT

http://www.mdp.ivv.nasa.gov

Data Set 2: The KC1 data set consists of 43 KLOC of C++ code. It consists of is 2107 observations. These metrics are class level metrics. They are given in Table III.

TABLE III
THE SEVENTEEN INDEPENDENT VARIABLES USED IN KC1 DATASET

NO.	VARIABLE
1	BRANCH _COUNT
2	CYCLOMATIC_COMPLEXITY
3	DESIGN_COMPLEXITY
4	LOC_TOTAL
5	ESSENTIAL_COMPLEXITY
6	HALSTEAD_CONTENT
7	HALSTEAD_DIFFICULTY
8	HALSTEAD_EFFORT
9	HALSTEAD_ERROR_EST
10	HALSTEAD_LENGTH
11	HALSTEAD_LEVEL
12	HALSTEAD_PROG_TIME
13	HALSTEAD_VOLUME
14	NUM_OPERANDS
15	NUM_OPERATORS
16	NUM_UNIQUE_OPERANDS
17	NUM_UNIQUE_OPERATORS

http://www.mdp.ivv.nasa.gov

III. ANALYSIS

Three samples from each data set were used. The first dataset had 10% missing data for the dependent variable fault proneness. Second and third data set with 20% and 30% missing data respectively as shown in the Figure1. All the samples were treated with the different MDT's and the results were compared. In LD the tuples with missing values were removed. So the accuracy achieved, in that case is minimum when compared to other MDT's. Comparison and accuracy of other MDT's are shown in the figures given below. Fig. 1 shows the accuracy of various techniques in both the data sets. In our analysis, we found that in KC4, where the data size is considerably small, using LD would further decrease the size of the data set and lead to biased results. MS and Interpolation produced inefficient results as compared to Regression and EM. In KC1, where the sample size is large, MS and Interpolation also gave efficient results as compared with Regression and EM. This is probably due to the pattern of the dependent values. The accuracy of the techniques has been summarised in Fig. 1.

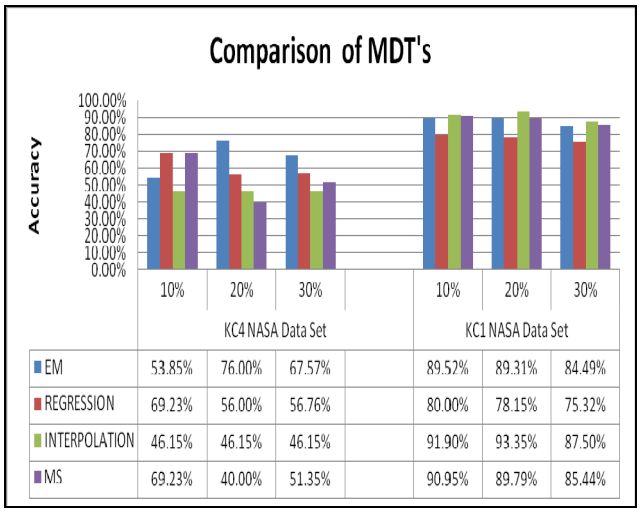


Fig. 1 Accuracy of MDT's over missing data

Fig. 2 shows the trends of various techniques in both the datasets.

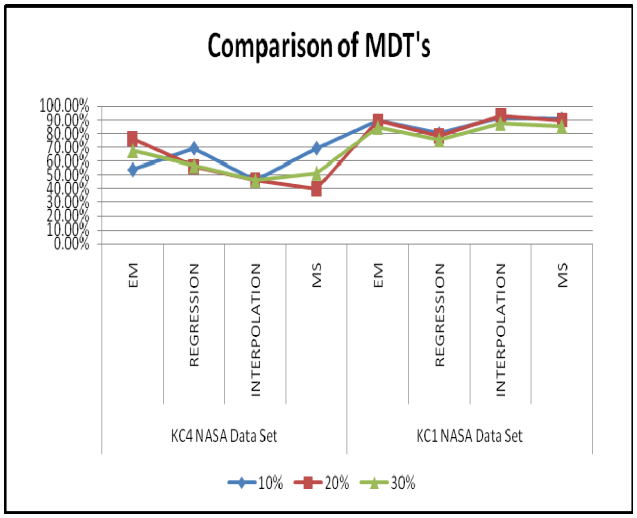


Fig. 2 Trends of Missing Data Techniques in KC4 and KC1 Data Sets

IV. Results and Discussions

The fault proneness for missing classes was predicted based on various Imputation techniques. The summary results show that all the techniques had their own strengths and weaknesses. Although, in our study, where the data size was small, results have shown that Listwise Deletion, Mean Substitution and Interpolation are less effective. They are less accurate as compared to Regression with an error term and EM Imputation techniques. In small data sets, using LD decrease the number of observations further and can result in biased results. So, this method is not recommended where the data size is small and also where the numbers of missing observations are more. In the case of KC1 where the data set was large all the techniques excluding LD produced marginally better results. The result was probably due to the pattern and proportion of missing data. On the whole, in both the data sets besides the number of observations and percentage of missingness, Expectation Maximization was the most appropriate approach for handling missing data of all sizes and proportions. Though, missing data can seriously affect our analysis and can lead to inaccurate results, still it is not just sufficient to simply apply missing value analysis and fill in a value using imputation techniques. The uncertainty of missing data and nature of variables should be clearly understood. The proportion, mechanism and response of missing data should be analysed before the selection of missing data technique used for prediction. Better estimates can be achieved by using Multiple Imputation techniques or by using a model based approach.

References

[1] R.J.A Little, D.B. Rubin, Statistical Analysis with missing data, *Wiley*, New York, 1987.
[2] D.B.Rubin, Multiple imputation for non response in surveys, *Wiley*, New York, 1987.
[3] J.Schafer, Analysis of incomplete multivariate data: Chapman and Hall, 1997.
[4] F.Harrell,"Regression modelling Strategies: With Applications to Linear Models, Logistic Regression, and Survival Analysis" Springer, New York, 2001.
[5] P.D. Allison, Missing Data, SAGE Publication, Inc, 2001..
[6] C.M. Musil, C.B.Warner, P.K.Yobas, and S.L. Jones, "A Comparison of Imputation Techniques for handling missing data," *Western Journal of Nursing Research*, vol.24, no. 5,pp.815-829, 2002.
[7] E.G. Johnson, "Considerations and techniques for the analysis of NAEP data," *Journal of Educational Statistics*, vol.14, pp.303-334,1989.
[8] C.J.Kaufman, "The application of logical imputation to household measurement", *Journal of the Market Research Society*, vol.30, pp.453-466, 1989.
[9] I.Myrtveit, E. Stensrud, and U.Olsson, "Analyzing Data Sets with missing Data: An Empirical Evaluation of Imputation Methods and Likelihood-Based Methods," *IEEE Transactions on Software Engineering*, vol.27, no.11, pp.1999-1013, 2001.
[10] K.Strike, K.E.El-Emam, N.Madhavji, "Software Cost Estimation with Incomplete Data," *IEEE Transactions on Software Engineering*, vol.27, no.10,890-908, 2001.R. W. Lucky, "Automatic equalization for digital communication," *Bell Syst. Tech. J.*, vol. 44, no. 4, pp. 547–588, Apr. 1965.
[11] M.Cartwright, M.J.Shepperd, and Q.Song, "Dealing with Missing Software Project data," In *Proc. of the 9th Int. Symp. on Software Metrics*, pp.154-165, 2003.
[12] B.Twala, M.Cartwright, M.J. Shepperd, "Ensemble of Missing Data Techniques to Improve Software Prediction Accracy," *ICSE*'06, 2006.
[13] B.Twala, "An Empirical Comparison of Techniques for handling Incomplete Data using Decision Trees," *Journal of Applied Artificial Intelligence*, vol.23, no. 5, pp.373-405, 2009.
[14] www.mdp.ivv.nasa.gov, NASA Metrics data Repository.

Open Science Index, Computer and Information Engineering Vol:4, No:2, 2010 publications.waset.org/9138/pdf