MULTIFUNCTIONAL OPTIMIZED GROUP METHOD DATA HANDLING FOR SOFTWARE EFFORT ESTIMATION

SITI HAJAR BINTI ARBAIN

A thesis submitted in fulfilment of the
requirements for the award of the degree of
Doctor of Philosophy

School of Computing
Faculty of Engineering
Universiti Teknologi Malaysia

JULY 2022

ACKNOWLEDGEMENT

In preparing this thesis, I was in contact with many people, researchers, academicians, and practitioners. They have contributed towards my understanding and thoughts. I wish to express my sincere appreciation to my main thesis supervisor, Dr Noorfa Haszlinna Mustaffa, for encouragement, guidance, critics, and friendship. I am also very thankful to my co-supervisor Professor Ts. Dr Dayang Norhayati bte Abang Jawawi and Dr Nor Azizah Ali for their guidance, advice, and motivation. Without their continued support and interest, this thesis would not have been the same as presented here. To my lovely husband, Khairul Ajwad Ab Samad, my family and my in-laws, thanks for supporting me emotionally and financially, also encouraging me in all of my pursuits and inspiring me to follow my dreams.

I am also indebted to Kementerian Pengajian Tinggi (KPT) for funding my PhD study. Librarians at UTM, UTAR, UTHM and UPSI also deserve special thanks for their assistance in supplying the relevant literatures.

My fellow postgraduate student should also be recognised for their support. My sincere appreciation also extends to all my colleagues, Nini, Kak Farah, Kak Su, Kak Jana, Hamizah, Aziran, Cla, Asiah and others who have helped at various occasions. Their views and tips are useful indeed. Unfortunately, it is not possible to list all of them in this limited space. I am grateful to all my family members and friends.

ABSTRACT

Nowadays, the trend of significant effort estimations is in demand. Due to its popularity, the stakeholder needs effective and efficient software development processes with the best estimation and accuracy to suit all data types. Nevertheless, finding the best effort estimation model with good accuracy is hard to serve this purpose. Group Method of Data Handling (GMDH) algorithms have been widely used for modelling and identifying complex systems and potentially applied in software effort estimation. However, there is limited study to determine the best architecture and optimal weight coefficients of the transfer function for the GMDH model. This study aimed to propose a hybrid multifunctional GMDH with Artificial Bee Colony (GMDH-ABC) based on a combination of four individual GMDH models, namely, GMDH-Polynomial, GMDH-Sigmoid, GMDH-Radial Basis Function, and GMDH-Tangent. The best GMDH architecture is determined based on L9 Taguchi orthogonal array. Five datasets (i.e., Cocomo, Dershanais, Albrecht, Kemerer and ISBSG) were used to validate the proposed models. The missing values in the dataset are imputed by the developed MissForest Multiple imputation method (MFMI). The Mean Absolute Percentage Error (MAPE) was used as performance measurement. The result showed that the GMDH-ABC model outperformed the individual GMDH by more than 50% improvement compared to standard conventional GMDH models and the benchmark ANN model in all datasets. The Cocomo dataset improved by 49% compared to the conventional GMDH-LSM. Improvements of 71%, 63%, 67%, and 82% in accuracy were obtained for the Dershanis dataset, Albrecht dataset, Kemerer dataset, and ISBSG dataset, respectively, as compared with the conventional GMDH-LSM. The results indicated that the proposed GMDH-ABC model has the ability to achieve higher accuracy in software effort estimation.

ABSTRAK

Pada masa kini, trend anggaran usaha yang ketara mendapat permintaan yang tinggi. Disebabkan popularitinya, pihak berkepentingan memerlukan proses pembangunan perisian yang berkesan dan cekap dengan anggaran dan ketepatan yang baik untuk disesuaikan dengan semua jenis data. Namun begitu, mencari model anggaran usaha terbaik dengan ketepatan yang baik adalah sukar untuk memenuhi tujuan ini. Algoritma Kumpulan Kaedah Pengendalian Data (GMDH) telah digunakan secara meluas untuk pemodelan dan mengenal pasti sistem yang kompleks dan berpotensi untuk digunakan dalam anggaran usaha perisian. Walau bagaimanapun, terdapat kajian yang terhad untuk menentukan seni bina terbaik dan pekali berat yang optimum bagi fungsi pemindahan untuk model GMDH. Kajian ini bertujuan untuk mencadangkan hibrid pelbagai fungsi GMDH dengan teknik Koloni Lebah Buatan (GMDH-ABC) berdasarkan gabungan empat model GMDH individu iaitu, GMDH-Polynomial, GMDH-Sigmoid, GMDH-Radial Basis Function, dan GMDH-Tangent. Seni bina GMDH terbaik ditentukan berdasarkan tatasusunan ortogonal L9 Taguchi. Lima set data (iaitu, Cocomo, Dershanais, Albrecht, Kemerer dan ISBSG) telah digunakan untuk mengesahkan model yang dicadangkan. Nilai yang hilang dalam set data digantikan dengan kaedah imputasi Berbilang MissForest (MFMI) yang dibangunkan. Ralat Peratusan Mutlak Minimum (MAPE) digunakan sebagai ukuran prestasi. Hasil kajian menunjukkan bahawa model GMDH-ABC mengatasi prestasi GMDH individu dengan peningkatan lebih daripada 50% berbanding model konvensional piawai GMDH dan model penanda aras iaitu ANN dalam semua set data. Set data Cocomo bertambah baik sebanyak 49% berbanding konvensional GMDH-LSM. Peningkatan sebanyak 71%, 63%, 67%, dan 82% dalam ketepatan telah dicapai untuk dataset Dershanis, dataset Albrecht, dataset Kemerer, dan dataset ISBSG masing-masing berbanding konvensional GMDH-LSM. Keputusan menunjukkan bahawa model GMDH-ABC yang dicadangkan mempunyai keupayaan untuk mencapai ketepatan yang lebih tinggi dalam anggaran usaha perisian.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ABC	-	Artificial Bee Colony
ANN	-	Artificial Neural Network
FPA	-	Function Point Analysis
GMDH	-	Group Method Data Handling
MAR	-	Missing At Random
MCAR	-	Missing Completely At Random
MFMI	-	MissForest Multiple Imputation
ML	-	Machine Learning
MNAR	-	Missing Not At Random
MR	-	Multiple Regression
SEE	-	Software Effort Estimation
TF	-	Transfer Function

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Overview

The software development process is growing increasingly complex these days. Software project development is one of the processes in the planning of software project management. Each activity under software project development should be updated on a frequent basis, depending on the size of the organisation and the work at hand. As a result, more sophisticated ways are required to handle several difficult challenges in this domain. Software effort estimation is one of the courses under software project development. It must be monitored by the manager in order to develop high-quality software at a reasonable cost within the time and budget constraints (Garbajosa, 2008; Ali and Gravino, 2021; Bakici *et al.*, 2021; For, 2021). The accuracy of software effort estimation is one of the important targets to propose any model development.

The complexities of estimations in the early stage of the software development process have raised concern for most developers. The uncertainty and unbalanced data cause a software effort estimation to be a challenging task especially in term of accuracy estimation. The challenge will increase as the size of the software project grows. Estimation accuracy errors can cost a lot of money in terms of project resources (Sinha and Gora, 2021). Hence, this study is proposed to investigate the use of several techniques to improve these challenging issues, focusing not only on the software effort engineering approach but also incorporating other methods such as imputing best techniques of missing data and the hybridization of the models, which can contribute to the enhancement of effort estimation accuracy because software development problems have many dimensions.

This chapter presents an introduction to the research work that will summarize all research content in this thesis. It describes research overview on researcher's motivation on this study to the enhancement of the accuracy of software effort estimation using heuristic deep learning GMDH techniques. The first section explains the background of the problem, followed by the statement of the problem, research questions, research objectives and scope of this research. The last part presents the significance of this study and structure of this thesis.

1.2 Background of the problem

The primary goal of the software development industry is to produce high-quality software at a reasonable cost under conditions. There is a lot of work to be done in the early stages of a software development project. Software effort estimation is one of the processes in a software development project, with the goal of producing high-quality accurate software that is delivered on time, on budget, and meets all the project's requirements. It is also known as an integral aspect of software engineering, with a focus on how to manage limited financial resources in a way that will enable the project to fulfil its goals in terms of time, cost, and size. Accurate software effort estimation is essential for successful planning, controlling, and completing software projects on time and within budget. The primary issues for upcoming software development are overestimation and underestimating, therefore there is a continuous need for accuracy in software effort estimation (SEE) model development.

In the standard software engineering economic study, different methods for effort estimation have been evaluated and used either in algorithmic or non-algorithmic approach. To accomplish software estimating, the algorithmic method is created using certain typical numerical demonstrating. The Constructive Cost Model (COCOMO), Function Point Analysis (FPA) model, Putnam Model are some examples of famous algorithmic approaches that have been used (Sinha and Gora, 2021). Another one is a non-algorithmic method in which the software project estimation uses an expert and previous software development experiences. In this

process, estimators must know about earlier completed projects similar to current projects (Iqbal *et al.*, 2017; Dutta *et al.*, 2019; Mahmood *et al.*, 2022).

Currently, due to advent of new technology, most researchers seem to be in competition with each other to provide a good method that can improve a conventional approach in software effort estimation process. In practice, the overall process of this effort estimation should be the focus in early steps, such as how the data should be prepared before it is analysed and how far the techniques used will enhance the features of the model used. The problem occurred when most of the researchers who require historical data simply assume or jump into an analytical process to develop an effort estimation model (Brown et al., 2018). However, considerably less research has been carried out on the pre-processing data in which to prepare a quality data before the data needs to be analysed (Somasundaram and Nedunchezhian, 2011; Alasadi and Bhaya, 2017). The quality of data is a significant concept that deals with various perspectives, of which the completeness is one of them. The question what data should be used seems not very important compared to which dataset is more reliable to be chosen. The researchers who study effort estimation should be warned about the problems caused by unbalanced datasets which can give bad impact on the model produced especially in term of accuracy model (Kitchenham, 1998; Dagliati et al., 2018; Zhang et al., 2018). Hence, to achieve such objective, the pre-processing imputation missing data techniques are an important consideration.

According to studies, both formal and expert techniques have their own limitations, and there is still a need to enhance the optimal one (Menzies et al., 2015). The effectiveness of processes is determined by the context data, methodologies applied, and the issue domain to be solved (Molokken and Jorgensen, 2003; Song et al., 2008; Jørgensen et al., 2009; Padmaja and Haritha, 2018). Machine learning (ML) and optimization model have been one of the popular techniques applied in domain of software effort estimation. The development of machine learning techniques in software effort estimation has been dealt with by current researchers for the past few decades. For example Artificial Neural Network (ANN) in combination with optimization process was conducted by many researchers as it was

useful to increase the accuracy of model by conducting a repeated cycle of its training data (Anifowose et al., 2017; Kumar et al., 2020a; Saikia et al., 2020). However, there is concern in application of ANN due to its inappropriate variable input selection, as there are many parameters that are hard to be tuned, with the result that most of the implementations of ANN are done on trial-and-error basis (Zatarain Cabada et al., 2020). Otherwise, the Taguchi method has been employed with a great success in experimental designs for problems with dealing multiple parameters caused by trial-and-error basis. Integrated of Taguchi with finding the optimal parameter design should be considered.

One sub-model of neural network is a Group Method of Data Handling (GMDH) algorithm which was first developed by Ivakhnenko (1971) for modelling and identification of complex systems. The GMDH model is known as a self-organizing heuristic modelling approach which began to attract the attention of the researchers compared to ANN (Kumar *et al.*, 2020b). It is very effective for solving modelling problems involving multiple input to single output data. Even though the GMDH model has been used in many domains of modelling, it has received little attention as an impact for software effort estimation (Ivakhnenko, 1971; Lee, 2015; Madala and Ivakhnenko, 2019; Malekzadeh *et al.*, 2019). In this study is intended to explore the integrated of GMDH model with Taguchi in design of experimental process and no more trial-and-error basis used.

Incorporating software effort estimation with a novel heuristic machine learning algorithm, on the other hand, is a popular research topic these days because it provides additional benefits such as the ability to enhance performance in feature selection and learn from previously collected data, which is primarily focused on predictive accuracy. Most of the researchers have implemented machine learning models to improve the significance of software effort estimation, but the accuracy of model has been questionable until now (Kumar and Srinivas, 2021). Thus, selecting the best features and model for software effort estimation is still an active domain among researchers (Erhan et al., 2020; Kumar and Venkatesan, 2020; Varshini and Kumari, 2020). Through the current investigations, improvement of software effort estimation is focused more on hyper-heuristics and multi-function methods to

represent the process of estimation model towards accurate model estimations. In order to alleviate the problem with basis GMDH model, numerous researchers have incorporated some of GMDH with other features models. According to Jirina (1994), as the complexity of the model increases, the degeneration of GMDH's accuracy could be due to the polynomial transfer function which causes multilayerness error to occur in GMDH's network. Meanwhile, Ivakhnenko also mentioned that the low accuracy in GMDH might be owing to the insufficient functional variety of the model (1985). Over the years, Kondo has applied several transfer functions in GMDH such as polynomial, logistic sigmoid and radial basis function (RBF) as seen in his works (Kondo et al., 1999; Kondo, 2002; Kondo and Ueno, 2009; Takao et al., 2017, 2018). According to Kondo (2003), employing heterogenous transfer functions within a model gives better results than using homogenous transfer function and it can fit the complexity of the nonlinear system.

In summary, the major goal in software effort estimation domain is to achieve accurate software effort estimations. The failure of the effort estimation is one of the reasons for researcher's intent to explore more on conventional study. There are many discussions towards software effort estimation accuracy and currently most of the researchers provide optimal artificial Neural Network model to deal with estimation accuracy (Dutta *et al.*, 2019; Kumar and Srinivas, 2021; Rao and Rao, 2021; Sharma and Vijayvargiya, 2021). However, the weakness of current neural network existing research had problems in the selection of the best architecture (Lin *et al.*, 2011), does not perform well in some of the datasets (Dan, 2013), the blackbox nature of neural network approach (Ughi *et al.*, 2021) and the accuracy of effort estimation still being questionable till now (Kumar and Srinivas, 2021; Mahmood *et al.*, 2022). To overcome the limitations of ANN, most of the researchers have provide one of the subset models of ANN that provide robust techniques, known as Group Method Data Handling (GMDH). It is however still exist some limitations of GMDH that need to be overcome such as the unstable optimal and internal parameter inside GMDH algorithm (Taušer and Buryan, 2011) and the low accuracy with insufficient functional variety of the model (Ugrasen *et al.*, 2014). This issues of GMDH have been highlighted among the researchers because it might affect the accuracy of the estimation model. In addition, the usage of multifunction GMDH has been quite popular among the researchers in finding the optimal GMDH model,

however there is lack of discussion toward the combinations of multifunction GMDH using optimization techniques. In other hand, more study has shown the weakness and inaccurate provision of estimations of software project in which the inconsistency, model complexity, unbalanced datasets have affected the performance and accuracy of the estimations. Each problem and data have their unique characteristics which cannot be used directly for every model even though that model is well known and already established. It is getting worse for a estimation model if the data provided is not clean and might affect the complexity and accuracy of the model (Ivakhnenko *et al.*, 2003).

1.3 Statement of the problem

The issues in software effort estimation are critical. The implementation of machine learning techniques in improving the effort estimation accuracy which focus on imputation of missing data are very important. The most important part of software effort estimation is finding the most accurate estimation accuracy which focuses on the process of development estimation model. The software project manager should decide which methodology is the best to avoid poor estimation.

The recent developments in variable selection methods have addressed the problems from the point of view of improving the performance of predictor's selection. It is noticed that the original data that was collected from online and certain datasets have some imperfect characteristics that need to undergo process of pre-processing treatment of missing data before proceeding to the next method procedure. Some recent studies have presented the awareness of the importance of treating missing data to improve effort estimation consistency (Twala *et al.*, 2005; Lang and Little, 2018; Carpenter and Smuk, 2021). Hence, pre-processing data is one of the most important methods before doing the next step procedures where it will handle the imperfect characteristic data such as missing value of data. Like any certifiable dataset, deficient or missing information is unavoidable. Missing values result in less effective imputation, which will decrease the accuracy of estimate model. Thus, treatment of missing values attribution is compulsory. The

completeness and quality of data are more precise than those inferences data analysis made from incomplete data.

There are several alternative ways of dealing with missing data. In some cases, deletion or elimination the missing variable is the default method for most procedures (Suguna and Thanushkodi, 2011; Rani and Solanki, 2021). However, handling missing data has received little attention in software engineering research and poor performance of estimation was expected as these techniques drastically reduce the sample size by eliminating a large amount of important sample set of data. There should be an empirical investigation of the robustness and accuracy of handling missing data. Most current techniques also used single imputations rather than multiple imputation. Single imputations, including both classical and modern methods, is generally simple with the purpose only to treat missing data; however, in multiple imputations, several sets of processes provided in between imputations make it more advantageous than single imputations (Gómez-Carracedo *et al.*, 2014; Rani and Solanki, 2021).

Furthermore, the implementation of the best estimation model in dataset of software development needs an adjustment with the help of heuristic machine learning techniques to enhance the accuracy of the best model (Amazal and Idri, 2021; Kumar and Srinivas, 2021; Mahmood *et al.*, 2022). Machine learning (ML) is a method that learns from the pattern of historical data and mostly helps in estimation process. The job is always integrated with artificial intelligence such as pattern recognition, planning, prediction, etc. and it is used when human expertise is limited. In effort estimation, the human expertise is unable to achieve the maintainability, which is why it is useful to incorporate with artificial expertise to develop effective estimation model.

There are many implementations and improvements of effort estimation model using machine learning techniques to overcome some limitations of their estimation accuracy which is applicable for standard software development. Artificial Neural Network (ANN) is one of the regular techniques used in software effort estimation in machine learning approach (Saeed *et al.*, 2018). There are many

hybrid implementations to overcome of specific limitations of ANN. One of the quite active discussions among practitioners of ANN is their trial and error setting of parameter features in terms of number of inputs, neurons, and layers (Dan, 2013; Tailor *et al.*, 2014). Most practitioners will just use previous suggestions to start a feature setting, without looking at other potentially better approaches. Furthermore, GMDH is quite famous model that is comparable to ANN(Ugrasen *et al.*, 2014; Ebtehaj *et al.*, 2015; Yahya *et al.*, 2019). Same as ANN, one of the main issues in GMDH model is the setting of parameter, such as neuron and layers. The initial guessing trial-and-basis of the number of neurons in a layer are irrelevant, that's the reason one of the experimental analysis Taguchi, has been applied. The Taguchi will help the initial parameter design of GMDH and avoid repeating experiments. In addition, the GMDH has the strength of getting a better modelling capability by combining with other optimization soft computing techniques. However, additional modification will be made on GMDH model by implementation of various transfer functions to increase GMDH model estimation performance itself. The variety of function used in GMDH will enhance the accuracy GMDH model. Improving the parameter coefficient of GMDH itself and choice of transfer function has attracted attention for improvement of GMDH performance and their performance has been discussed in effort estimation domains. Previously, estimate the unknown coefficients in every layer of GMDH recurrently employed using least square methos (LSM). It is however, researchers found that the regression model increase the multicollinearity, resulting in unstable production of coefficient (Tauser,2011). Hence, the metaheuristic techniques such as Artificial Bee Colony (ABC) algorithm need to be hybrid with GMDH to improve the accuracy of software effort estimation.

In addition, an estimation model is no longer acceptable for only using old datasets and without looking at the improvement techniques used. The issue starts when there is a lack of support decision for a software project manager during effort estimation in term of accuracy estimation. When dealing with incompleteness of data, the limitations of GMDH as predictive model and decision of multiple best models, it is hard for project manager to decide which is the best model to propose. In existing research, most of the estimation model is develop without consider pre-processing missing data. The setting of parameter algorithm in traditional GMDH also doesn't have consistent techniques and most of it only consider individual effort

estimation model. The problems of these issues will impact to quality of data preparation and the conventional setting model not in optimal parameter. In addition, when dealing with inconsistency of different setting and model, how to choose the best among the models.

Therefore, the incorporation of these types of analysis should employ latest improvement of the algorithm, weight or neuron which result in minimum error between predicted and actual output. The enhancement of software effort estimation using heuristic hybrid GMDH model with an optimization technique, and the setting parameter of its early phases, will be carried out after imputation of missing data.

Figure 1.1 shows the summary of problem statement and knowledge gap as the purpose of study. Based on evaluation measures, datasets, and other relevant factors, academics and practitioners are attempting to determine which estimation technique produces more accurate results (Grimstad, 2005; Jørgensen and Grimstad, 2011; Bukhari and Malik, 2012; Mahmood *et al.*, 2022). In this research, there are three main issues that will be highlighted causes the lack of support decision for a software project manager during effort estimation planning in terms of accuracy, which is completeness of data preparation analysis, limitations of GMDH as predictive model, and how to decide when dealing with multiple best models. The current estimation model uses single approach, which causes problems in terms of unavailability to support estimation model, and inability to make suitable decisions.

As explained before, the issues in software effort estimation are critical. The implementation of machine learning techniques in improving the effort estimation accuracy which focus on imputation of missing data are very important. The three main consideration in this study, which is data quality preparation in terms of completeness of missing data, limitations of GMDH model, and dealing with multifunction of model, are the issues that hinder the achievement of developing accurate software effort estimation to support decision making of software project manager.

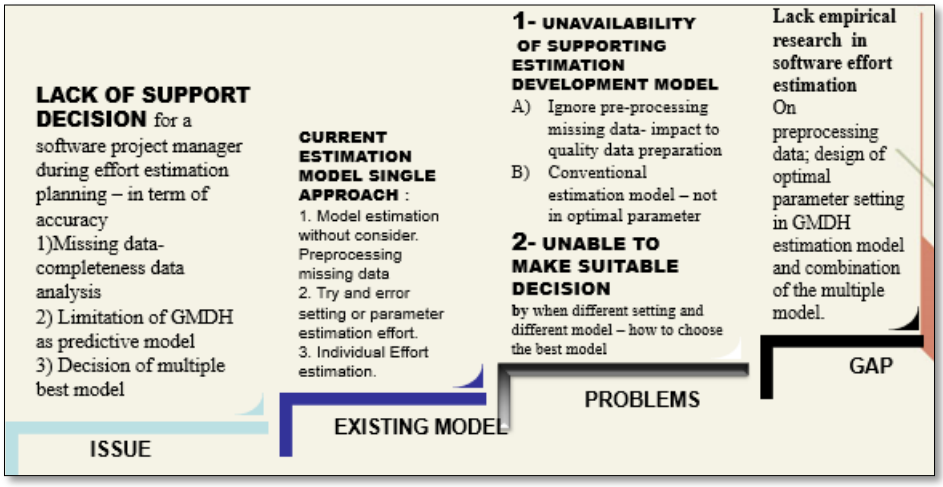

Figure 1.1 Problem Statement and Knowledge's Gap

1.4 Research Questions

Many techniques have been used to improve the estimation accuracy of heuristic models. However, most experts and formal methods have given less consideration to the problems in early preparation of the dataset studies and improvement of techniques incorporated in the model. The question should not only investigate the process of effort estimation model but will focus on the enhancement of machine learning algorithm, which is GMDH model that supports the reliability of model accuracy.

Our main research question is how to enhance the accuracy of GMDH model in estimate software effort. Based on the main research problem, three specific research questions have been formulated as:

1) How to impute missing data in software effort estimation while confronting various mechanism of missingness?
2) How to determine the optimal parameters setting for GMDH?
3) How can hybrid multifunction GMDH with metaheuristic technique improve software effort estimation accuracy?

1.5 Research Objective

The objective of this research has been formulated based on the research questions, which are listed as follows:

1) To propose missing data imputation model based on MissForest Multiple Imputation approach in order to improve the quality of dataset for software effort estimation.
2) To design GMDH architecture based on Taguchi method to obtain the optimal parameter setting of GMDH.
3) To propose Hybrid GMDH-ABC model based on weighted-based combination optimization to enhance accuracy software effort estimation.

1.6 Scope of research

1. This study focuses more on issues of enhancement of accuracy in software effort estimation using enhancement GMDH techniques.
2. The target is to analyse all the factors that influence the accuracy of estimation model including the quality of data preparation, design of experiments in estimation model and modified algorithm of GMDH used.
3. The performance of the proposed model will be tested on several completed data software project datasets taken from online resources and International Software Benchmarking Standard Group (ISBSG), which consist of largest multi-organization software engineering repository data for academic research purposes and considering another hybrid techniques (PABE, ABE) and ANN as benchmark of standard software effort estimation techniques.
4. The software used in this research are:-

   Matlab : analyse the performance of GMDH and hybrid GMDH ABC
   Pyton : Imputation of missing data – Missforest Imputation
   SPSS : analyse the performance of Taguchi

1.7 Significances of research

This study has universal importance in this early phase of its implementation. These ordinary significations are based on the literature and brainstorm thinking, and on knowing the real significances of the study, which will be required to accomplish the objective of the study. It will contribute to the research on effort estimation by developing an efficient model that allows software manager to determine the best effort model to estimate software project. It is also intended to help software practitioners to reduce costing.

Financial of many organizations nowadays are being affected by investment in software and their effort estimation. Due to lack of proper tool and method, doing software effort estimation in early stage has become difficult, tedious, time-consuming and error prone. Software project managers need mechanisms to understand and resolve estimation task, which involves not only using their expert judgment but also a new mechanism approach. Providing practical and improvement of effort estimation models are the most complex activity stage.

Therefore, software effort estimation is investable and most practitioners still receive new requirements while the software is continuously evolving and rapidly needed in most industries. Based on the explanation above, it is proven that estimation accuracy is the primary objective of each software engineering community. The researcher may significantly reduce the effort estimation depending purely upon the amount of reliable information available about the software that needs to be developed. An important research area in software effort estimation is the further exploration of heuristic machine learning techniques to provide the best estimation model. It can help software project manager manage the performance in planning the software effort estimation task, especially for Malaysia organizations.

1.8 Thesis Organization

This report builds on some research studies that have previously been reported in conference papers, journal paper, and a book chapter. This section provides a brief description that has been converted into chapters.

Chapter 1 illustrates an introduction and brief overview of researchers including the background of research, formulation problems, objectives, scopes, the significance of the study, and report organizations.

Chapter 2 it discusses literature reviews that discuss the related theories to be used as the foundation for this study. Here, the summaries of previous techniques in domain software effort estimation, GMDH approach and pre-processing techniques will be explored.

Chapter 3 will cover the methodology and operational framework of studies to achieve the objectives of this research. Since three objectives have been formulated, four phases of methodology will be explained in detail here. Design and Development Research (DDR) approach was used in this study to produce an heuristic model in software effort estimation.

Chapter 4 discusses the phase of pre-processing data analysis. The choice of data, mechanism, and imputations of missing data will be explained here. Also, the data division and early Taguchi GMDH setting will be detailed here.

Chapter 5 will briefly discuss the second phase, which is development of four individual function GMDH models in software effort estimation. The Taguchi are used to start the experiment design in each development model.

Chapter 6 will then briefly discuss the findings of the incorporation of ABC techniques to replace the conventional Least square method in estimate coefficient inside GMDH. Also will discuss a hybrid Multifunction GMDH-ABC.

Chapter 7 will discuss all the conclusions of the findings and suggest research directions for future researchers.

REFERENCES

Abdelali, Z., Mustapha, H. and Abdelwahed, N. (2019) 'Investigating the use of random forest in software effort estimation', *Procedia Computer Science*. Elsevier B.V., 148, pp. 343–352.

Alade, O. A., Sallehuddin, R., Haizan, N., Radzi, M. and Selamat, A. (2020) 'Missing Data Characteristics and the Choice of Imputation Technique : An Empirical Study Missing Data Characteristics and the Choice of Imputation Technique : An Empirical Study', (May).

Alasadi, S. A. and Bhaya, W. S. (2017) 'Review of data preprocessing techniques in data mining', *Journal of Engineering and Applied Sciences*, 12(16), pp. 4102–4107.

Ali, A. and Gravino, C. (2021) 'Improving software effort estimation using bio-inspired algorithms to select relevant features: An empirical study', *Science of Computer Programming*.

Alsaber, A. R., Pan, J. and Al-Hurban, A. (2021) 'Handling complex missing data using random forest approach for an air quality monitoring dataset: A case study of kuwait environmental data (2012 to 2018)', *International Journal of Environmental Research and Public Health*, 18(3), pp. 1–26.

Amazal, F. A. and Idri, A. (2021) 'Estimating software development effort using fuzzy clustering-based analogy', *Journal of Software: Evolution and Process*, 33(4), pp. 1–23.

Amiri, M. and Soleimani, S. (2021) 'ML-based group method of data handling: an improvement on the conventional GMDH', *Complex & Intelligent Systems*. Springer International Publishing, 7(6), pp. 2949–2960.

Andrew, B. and Selamat, A. (2012) 'Systematic Literature Review of Missing Data Imputation Techniques for Effort Prediction', *International Proceedings of Computer Science & Information Technology*, 45(Icikm), pp. 222–226.

Anifowose, F. A., Labadin, J. and Abdulraheem, A. (2017) 'Hybrid intelligent systems in petroleum reservoir characterization and modeling: the journey so far and the challenges ahead', *Journal of Petroleum Exploration and Production Technology*. Springer Berlin Heidelberg, 7(1), pp. 251–263.

Bai, Y., Chen, M., Zhou, P., Zhao, T., Lee, J. D., Kakade, S., Wang, H. and Xiong, C. (2020) 'How Important is the Train-Validation Split in Meta-Learning?'

Bakici, T., Nemeh, A. and Hazir, O. (2021) 'Big Data Adoption in Project Management: Insights From French Organizations', *IEEE Transactions on Engineering Management*. IEEE, pp. 1–15.

Bannigan, P., Aldeghi, M., Bao, Z., Häse, F., Aspuru-Guzik, A. and Allen, C. (2021) 'Machine learning directed drug formulation development', *Advanced Drug Delivery Reviews*. Elsevier B.V., 175, p. 113806.

Beesley, L. J., Bondarenko, I., Elliot, M. R., Kurian, A. W., Katz, S. J. and Taylor, J. M. G. (2021) 'Multiple imputation with missing data indicators', *Statistical Methods in Medical Research*, 30(12), pp. 2685–2700.

Bilgaiyan, S., Mishra, S. and Das, M. (2016) 'A Review of Software Cost Estimation in Agile Software Development Using Soft Computing Techniques', *2016 2nd International Conference on Computational Intelligence and Networks (CINE)*, pp. 112–117.

Boehm, B., Valerdi, R. and Honour, E. (2008) 'The ROI of systems engineering: Some quantitative results for software-intensive systems', *Systems Engineering*, 11(3), pp. 221–234.

Boehm, B., Abts, C. and Chulani, S. (2000) 'Software development cost estimation approaches—A survey', *Annals of Software Engineering*, 10(1–4), pp. 177–205.

Borade, J. G. and Khalkar, V. R. (2013) 'Software Project Effort and Cost Estimation Techniques', *International Journal of Advanced Research in Computer Science and Software Engineering*, 3(8), pp. 730–739.

Bosu, M. F. and Macdonell, S. G. (2019) 'Experience : Quality Benchmarking of Datasets Used', 11(4).

Brown, A. W., Kaiser, K. A. and Allison, D. B. (2018) 'Issues with data and analyses: Errors, underlying themes, and potential solutions', *Proceedings of the National Academy of Sciences of the United States of America*, 115(11), pp. 2563–2570.

Bukhari, S. and Malik, A. A. (2012) 'Determining the factors affecting the accuracy of effort estimates for different application and task types', *Proceedings - 10th International Conference on Frontiers of Information Technology, FIT 2012*, pp. 41–45.

van Buuren, S. and Groothuis-Oudshoorn, K. (2011) ‘mice: Multivariate imputation by chained equations in R’, *Journal of Statistical Software*, 45(3), pp. 1–67.

Calabrese, B. (2018) ‘Data cleaning’, in *Encyclopedia of Bioinformatics and Computational Biology: ABC of Bioinformatics*.

Carpenter, J. R. and Kenward, M. G. (2012) *Multiple Imputation and its Application*, *Multiple Imputation and its Application*.

Carpenter, J. R. and Smuk, M. (2021) ‘Missing data: A statistical framework for practice’, *Biometrical Journal*, 63(5), pp. 915–947.

Choudhary, K. (2010) ‘GA Based Optimization of Software Development Effort Estimation’, *International Journal Of Computer Science And …*, 8491(L), pp. 38–40.

Dagliati, A., Marini, S., Sacchi, L., Cogni, G., Teliti, M., Tibollo, V., De Cata, P., Chiovato, L. and Bellazzi, R. (2018) ‘Machine Learning Methods to Predict Diabetes Complications’, *Journal of Diabetes Science and Technology*, 12(2), pp. 295–302.

Dan, Z. (2013) ‘Improving the accuracy in software effort estimation: Using artificial neural network model based on particle swarm optimization’, *Proceedings of 2013 IEEE International Conference on Service Operations and Logistics, and Informatics*, pp. 180–185.

Dean, E. B. and Unal, R. (1991) ‘Taguchi Approach to Design Optimization for Quality and Cost: An Overview’, *Annual Conference of the International Society of Parametric Analysts*, pp. 1–10.

Demirtas, H. (2005) ‘Multiple imputation under Bayesianly smoothed pattern-mixture models for non-ignorable drop-out’, *Statistics in Medicine*, 24(15), pp. 2345–2363.

Dutta, K., Gupta, V. and Dave, V. S. (2019) ‘Analysis and comparison of neural network models for software development effort estimation’, *Journal of Cases on Information Technology*, 21(2), pp. 88–112.

Ebtehaj, I., Bonakdari, H., Zaji, A. H., Azimi, H. and Khoshbin, F. (2015) ‘GMDH-type neural network approach for modeling the discharge coefficient of rectangular sharp-crested side weirs’, *Engineering Science and Technology, an International Journal*. Elsevier Ltd, 18(4), pp. 746–757.

Erhan, D. D., Tarhan, A. K. and Özakıncı, R. (2020) ‘Selecting suitable software effort estimation method’, *CEUR Workshop Proceedings*, 2725, pp. 1–16.

Farlow, S. J. (1981) ‘The GMDH Algorithm of Ivakhnenko’, *The American Statistician*.

For, O. (2021) ‘Software Quality in Report’, pp. 1–46.

Garbajosa, J. (2008) ‘The emerging ISO International Standard for Certification of Software Engineering Professionals’, in *IFIP International Federation for Information Processing*.

Gómez-Carracedo, M. P., Andrade, J. M., López-Mahía, P., Muniategui, S. and Prada, D. (2014) ‘A practical comparison of single and multiple imputation methods to handle complex missing data in air quality datasets’, *Chemometrics and Intelligent Laboratory Systems*. Elsevier B.V., 134, pp. 23–33.

Grimstad, S. (2005) ‘Understanding of estimation accuracy in software development projects’, *Proceedings - International Software Metrics Symposium*, 2005(Metrics), pp. 387–388.

Groenwold, R. H. H. and Dekkers, O. M. (2020) ‘Missing data: The impact of what is not there’, *European Journal of Endocrinology*, 183(4), pp. E7–E9.

Helenowski, I. B. (2015) ‘Advantages and Advancements of Multiple Imputation’, *Biometrics & Biostatistics International Journal*.

Ho, D., Capretz, L. F., Huang, X. and Ren, J. (2015) ‘Neuro-Fuzzy Algorithmic (NFA) Models and Tools for Estimation’, *arXiv:1508.00037 [cs]*, (October), pp. 1–5.

Huang, J., Li, Y. F., Keung, J. W., Yu, Y. T. and Chan, W. K. (2017) ‘An empirical analysis of three-stage data-preprocessing for analogy-based software effort estimation on the ISBSG data’, *Proceedings - 2017 IEEE International Conference on Software Quality, Reliability and Security, QRS 2017*, pp. 442–449.

Hughes, R. A., Heron, J., Sterne, J. A. C. and Tilling, K. (2019) ‘Accounting for missing data in statistical analyses: Multiple imputation is not always the answer’, *International Journal of Epidemiology*.

Ibrahim, R. (2011) ‘Demystifying the Arduous doctoral journey: The eagle vision of a research proposal’, *Electronic Journal of Business Research Methods*, 9(2), pp. 130–140.

Idri, A., Amazal, F. A. and Abran, A. (2016) ‘Accuracy Comparison of Analogy-Based Software Development Effort Estimation Techniques’, *International*

Journal of Intelligent Systems.

Iqbal, S. Z., Idrees, M., Sana, A. Bin and Khan, N. (2017) 'Comparative Analysis of Common Software Cost Estimation Modelling Techniques Comparative Analysis of Common Software Cost Estimation Modelling Techniques', (July).

Ivakhnenko, A. G. (1971) 'Polynomial Theory of Complex Systems', *IEEE Transactions on Systems, Man and Cybernetics*, 1(4), pp. 364–378.

Ivakhnenko, A. G., Savchenko, E. A. and Ivakhnenko, G. A. (2003) 'Problems of future GMDH algorithms development', *Systems Analysis Modelling Simulation*, 43(10), pp. 1301–1309.

Jahed Armaghani, D., Hasanipanah, M., Bakhshandeh Amnieh, H., Tien Bui, D., Mehrabi, P. and Khorami, M. (2020) 'Development of a novel hybrid intelligent model for solving engineering problems using GS-GMDH algorithm', *Engineering with Computers*.

Jørgensen, M., Boehm, B. and Rifkin, S. (2009) 'Software development effort estimation: Formal models or expert judgment?', *IEEE Software*, 26(2), pp. 14–19.

Jørgensen, M. and Grimstad, S. (2011) 'The impact of irrelevant and misleading information on software development effort estimates: A randomized controlled field experiment', *IEEE Transactions on Software Engineering*, 37(5), pp. 695–707.

Kang, H. (2013) 'The prevention and handling of the missing data', *Korean Journal of Anesthesiology*, 64(5), pp. 402–406.

Kaur, M. (2018) 'A fuzzy logic approach to software development effort estimation', pp. 125–127.

Khatibi, V. and Jawawi, D. N. . (2010) 'Software Cost Estimation Methods : A Review', *Journal of Emerging Trends in Computing and Information Sciences*.

Khoshgoftaar, T. M., Van Hulse, J. and Napolitano, A. (2011) 'Comparing boosting and bagging techniques with noisy and imbalanced data', *IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans*.

Kitchenham, B. (1998) 'A procedure for analyzing unbalanced datasets', *IEEE Transactions on Software Engineering*, 24(4), pp. 278–301.

Kumar, K. H. and Srinivas, K. (2021) 'Preliminary performance study of a brief

review on machine learning techniques for analogy based software effort estimation', *Journal of Ambient Intelligence and Humanized Computing*. Springer Berlin Heidelberg, (0123456789).

Kumar, P. S., Behera, H. S., K, A. K., Nayak, J. and Naik, B. (2020a) 'Advancement from neural networks to deep learning in software effort estimation : Perspective of two decades', *Computer Science Review*. Elsevier Inc., 38, p. 100288.

Kumar, P. S., Behera, H. S., K, A. K., Nayak, J. and Naik, B. (2020b) 'Advancement from neural networks to deep learning in software effort estimation : Perspective of two decades', *Computer Science Review*. Elsevier Inc., 38, p. 100288.

Kumar, S. and Venkatesan, P. R. (2020) 'Hyperparameters tuning of ensemble model for software effort estimation', *Journal of Ambient Intelligence and Humanized Computing*. Springer Berlin Heidelberg, (0123456789).

Kumari, S., Ali, M. and Pushkar, S. (2015) 'Fuzzy Clustering and Optimization Model for Software Cost Estimation', *International Journal of Engineering and Technology (IJET)*, 6(6).

Lang, K. M. and Little, T. D. (2018) 'Principled missing data treatments', *Prevention Science*. Prevention Science, 19(3), pp. 284–294.

Lee, K. (2015) 'Prediction of crack for drilling process on alumina using neural network and taguchi method', *Advances in Materials Science and Engineering*, 2015.

Leevy, J. L., Khoshgoftaar, T. M., Bauder, R. A. and Seliya, N. (2018) 'A survey on addressing high-class imbalance in big data', *Journal of Big Data*.

LEUNG, H. and FAN, Z. (2002) 'Software Cost Estimation', *Information and Software Technology*, 34(10), pp. 307–324.

Lin, J. C., Chang, C. T. and Huang, S. Y. (2011) 'Research on software effort estimation combined with genetic algorithm and support vector regression', in *Proceedings - 2011 International Symposium on Computer Science and Society, ISCCS 2011*.

Linton, J. D., Jiang, Q., Gatti, C. J. and Embrechts, M. J. (2013) 'Discussion of Kapsiz, M., Durat, M., Ficici, F. (2011). Friction and wear studies between cylinder liner and piston ring pair using Taguchi design method. Advances in Engineering Software, 42(8), 595-603', *Advances in Engineering Software*.

Little, R. J. A. (1988) ‘A test of Missing Completely at Random’, *Journal of American Statistical Association*, 83(404), pp. 1198–1202.

Madala, H. R. and Ivakhnenko, A. G. (2019) *Inductive Learning Algorithms for Complex Systems Modeling*, *Inductive Learning Algorithms for Complex Systems Modeling*.

Madhan Shridhar Phadke (1995) *Quality Engineering Using Robust Design: Phadke, Madhav S: 9780137451678: Amazon.com: Books*. Prentice Hall; Illustrated edition (May 22, 1989).

Mahmood, Y., Kama, N., Azmi, A., Khan, A. S. and Ali, M. (2022) ‘Software effort estimation accuracy prediction of machine learning techniques: A systematic performance evaluation’, *Software - Practice and Experience*, 52(1), pp. 39–65.

Mair, C., Shepperd, M. and Jørgensen, M. (2005) ‘An analysis of data sets used to train and validate cost prediction systems’, *ACM SIGSOFT Software Engineering Notes*, 30(4), pp. 1–6.

Maleki, I., Ghaffari, A. and Masdari, M. (2014) ‘A New Approach for Software Cost Estimation with Hybrid Genetic Algorithm and Ant Colony Optimization’, *International Journal of Innovation and Applied Studies*, 5(1), pp. 72–81.

Malekzadeh, M., Kardar, S. and Shabanlou, S. (2019) ‘Simulation of groundwater level using MODFLOW, extreme learning machine and Wavelet-Extreme Learning Machine models’, *Groundwater for Sustainable Development*. Elsevier B.V., 9, p. 100279.

Menzies, T., Kocagüneli, E., Minku, L., Peters, F. and Turhan, B. (2015) ‘How to Adapt Models in a Dynamic World’, *Sharing Data and Models in Software Engineering*, pp. 267–290.

Menzies, T., Yang, Y., Mathew, G., Boehm, B. and Hihn, J. (2016) ‘Negative results for software effort estimation’, *Empirical Software Engineering*. Empirical Software Engineering, pp. 1–26.

Molokken, K. and Jorgensen, M. (2003) ‘A review of software surveys on software effort estimation’, in *2003 International Symposium on Empirical Software Engineering, 2003. ISESE 2003. Proceedings.*, pp. 223–230.

Nathanael, E. H., Hendradjaya, B., Sunindyo, D. and Sc, M. (2015) ‘Study of Algorithmic Method and Model for Effort Estimation in Big Data Software Development Case Study : Geodatabase’, pp. 427–432.

Padmaja, M. and Haritha, D. (2017) ‘Software Effort Estimation Using Grey Relational Analysis’, *International Journal of Information Technology and Computer Science*, 9(5), pp. 52–60.

Padmaja, M. and Haritha, D. (2017) ‘Software Effort Estimation using Meta Heuristic Algorithm’, *International Journal of Advanced Research in Computer Science*, 8(5), pp. 196–201.

Padmaja, M. and Haritha, D. (2018) ‘Optimization of Process Parameters Using GreyTaguchi Method for Software Effort Estimation of Software Project’, *International Journal of Image, Graphics and Signal Processing*, 10(9), pp. 10–16.

Passos, E. B., Medeiros, D. B., Pedro, A. S. N. and Esteban, W. G. C. (2011) ‘Turning Real-World Software Development into a Game Challenge Punishment Reward Rules User Action’, *Brazilian Symposium on Games and Digital Entertainment*, pp. 1–8.

Qin, X. and Fang, M. (2011) ‘Summarization of software cost estimation’, in *Procedia Engineering*, pp. 3027–3031.

Rajper*, S. and and Zubair A. Shaikh (2016) ‘Software Development Cost Estimation Approaches - A Survey’, *Indian Journal of Science and Technology*, 9((31)), pp. 1–5.

Rani, S. and Solanki, A. (2021) ‘Data imputation in wireless sensor network using deep learning techniques’, in *Lecture Notes on Data Engineering and Communications Technologies*.

Rankovic, N., Rankovic, D., Ivanovic, M. and Lazic, L. (2021) ‘A new approach to software effort estimation using different Artificial Neural Network architectures and Taguchi Orthogonal Arrays’, *IEEE Access*, 9, pp. 1–1.

Rao, K. E. and Rao, G. A. (2021) ‘Ensemble learning with recursive feature elimination integrated software effort estimation: a novel approach’, *Evolutionary Intelligence*.

Roslan Armina1, Azlan Mohd Zain2, N. A. A. and R. S. (2019) ‘A review on missing data value estimation using imputation algorithm’, *Journal of Advanced Research in Dynamical and Control Systems*, 11(7 Special Issue), pp. 312–318.

Roth, A. and Xing, X. (1994) ‘Jumping the gun: Imperfections and institutions related to the timing of market transactions’, *American Economic Review*.

Saeed, A., Butt, W. H., Kazmi, F. and Arif, M. (2018) 'Survey of software development effort estimation techniques', *ACM International Conference Proceeding Series*, pp. 82–86.

Saikia, P., Baruah, R. D., Singh, S. K. and Chaudhuri, P. K. (2020) 'Artificial Neural Networks in the domain of reservoir characterization: A review from shallow to deep models', *Computers and Geosciences*. Elsevier Ltd, 135, p. 104357.

Schafer, J. L. and Graham, J. W. (2002) 'Missing data: Our view of the state of the art', *Psychological Methods*, 7(2), pp. 147–177.

Schafer, J. L. and Olsen, M. K. (1998) 'Multiple imputation for multivariate missing-data problems: A data analyst's perspective', *Multivariate Behavioral Research*, 33(4), pp. 545–571.

Shah, M. A., Jawawi, D. N. A., Isa, M. A., Wakil, K., Younas, M. and Mustafa, A. (2019) 'MINN: A missing data imputation technique for Analogy-Based Effort Estimation', *International Journal of Advanced Computer Science and Applications*, 10(2), pp. 222–232.

Shahavi, M. H., Hosseini, M., Jahanshahi, M., Meyer, R. L. and Darzi, G. N. (2016) 'Clove oil nanoemulsion as an effective antibacterial agent: Taguchi optimization method', *Desalination and Water Treatment*, 57(39), pp. 18379–18390.

Shahpar, Z., Bardsiri, V. K. and Bardsiri, A. K. (2021) 'Polynomial analogy-based software development effort estimation using combined particle swarm optimization and simulated annealing', *Concurrency and Computation: Practice and Experience*, 33(20).

Sharma, S. and Vijayvargiya, S. (2021) 'Applying Soft Computing Techniques for Software Project Effort Estimation Modelling', in *Lecture Notes in Electrical Engineering*.

Siddique, J. and Belin, T. R. (2008) 'Using an Approximate Bayesian Bootstrap to multiply impute nonignorable missing data', *Computational Statistics and Data Analysis*. Elsevier B.V., 53(2), pp. 405–415.

Sinha, R. R. and Gora, R. K. (2021) 'Software effort estimation using machine learning techniques', in *Lecture Notes in Networks and Systems*.

Smieja, M., Struski, Ł., Tabor, J., Zielinski, B. and Spurek, P. (2018) 'Processing of missing data by neural networks', in *Advances in Neural Information Processing Systems*.

Smith, B. I., Chimedza, C. and Bührmann, J. H. (2021) 'Random forest missing data imputation methods: Implications for predicting at-risk students', in *Advances in Intelligent Systems and Computing*.

Somasundaram, R. S. and Nedunchezhian, R. (2011) 'Evaluation of Three Simple Imputation Methods for Enhancing Preprocessing of Data with Missing Values', *International Journal of Computer Applications*, 21(10), pp. 14–19.

Song, Q., Shepperd, M., Chen, X. and Liu, J. (2008) 'Can k-NN imputation improve the performance of C4.5 with small software project data sets? A comparative evaluation', *Journal of Systems and Software*, 81(12), pp. 2361–2370.

Stekhoven, D. J. and Bühlmann, P. (2012) 'Missforest-Non-parametric missing value imputation for mixed-type data', *Bioinformatics*.

Strike, K., El Emam, K. and Madhavji, N. (2001) 'Software cost estimation with incomplete data', *IEEE Transactions on Software Engineering*, 27(10), pp. 890–908.

Stuart, E. A., Azur, M., Frangakis, C. and Leaf, P. (2009) 'Multiple imputation with large data sets: A case study of the children's mental health initiative', *American Journal of Epidemiology*, 169(9), pp. 1133–1139.

Suguna, N. and Thanushkodi, K. (2011) 'A weighted bee colony optimisation hybrid with rough set reduct algorithm for feature selection in the medical domain', *International Journal of Granular Computing, Rough Sets and Intelligent Systems*.

Tailor, O., Kumar, A. and Rijwani, P. (2014) 'a New High Performance Neural Network Model for Software Effort Estimation', 1(3).

Tang, F. and Ishwaran, H. (2017) 'Random forest missing data algorithms', *Statistical Analysis and Data Mining*, 10(6), pp. 363–377.

Taušer, J. and Buryan, P. (2011) 'Exchange rate predictions in international financial management by enhanced GMDH algorithm', *Prague Economic Papers*.

Thompson, R. L., Higgins, C. A. and Howell, J. M. (1991) 'Personal Computing: Toward a Conceptual Model of Utilization Utilization of Personal Computers Personal Computing: Toward a Conceptual Model of Utilization1', *Source: MIS Quarterly*, 15(1), pp. 125–143.

Tofallis, C. (2015) 'Erratum: A better measure of relative prediction accuracy for model selection and model estimation (Journal of the Operational Research Society (2015) 66:3 (524) DOI: 10.1057/jors.2014.103))', *Journal of the*

Operational Research Society, 66(3), p. 524.

Twala, B., Cartwright, M. and Shepperd, M. (2005) 'Comparison of various methods for handling incomplete data in software engineering databases', in *2005 International Symposium on Empirical Software Engineering, 2005.*, pp. 102–111.

Ughi, G., Abrol, V. and Tanner, J. (2021) 'An empirical study of derivative-free-optimization algorithms for targeted black-box attacks in deep neural networks', *Optimization and Engineering*. Springer US, (0123456789).

Ugrasen, G., Ravindra, H. V., Prakash, G. V. N. and Keshavamurthy, R. (2014) 'Estimation of Machining Performances Using MRA, GMDH and Artificial Neural Network in Wire EDM of EN-31', *Procedia Materials Science*. The Authors, 6(Icmpc), pp. 1788–1797.

Varshini, A. G. P. and Kumari, K. A. (2020) 'estimation : A review', (2), pp. 2094–2103.

Waljee, A. K., Mukherjee, A., Singal, A. G., Zhang, Y., Warren, J., Balis, U., Marrero, J., Zhu, J. and Higgins, P. D. R. (2013) 'Comparison of imputation methods for missing laboratory data in medicine', *BMJ Open*.

Watt, A. (2014) '2. Project Management Overview'. BCcampus.

Wijayasiriwardhane, T., Lai, R. and Kang, K. C. (2011) 'Effort estimation of component-based software development – a survey', *IET Software*, 5(2), p. 216.

Yahya, A. E., Samsudin, R. and Ilman, A. S. (2020) *A genetic algorithm-based grey model combined with fourier series for forecasting tourism arrivals in langkawi island malaysia*, *Advances in Intelligent Systems and Computing*.

Yahya, N. A., Samsudin, R., Shabri, A. and Saeed, F. (2019) 'Combined group method of data handling models using artificial bee colony algorithm in time series forecasting', *Procedia Computer Science*. Elsevier B.V., 163, pp. 319–329.

Yang, S. (2019) 'Flexible Imputation of Missing Data, 2nd ed.', *Journal of the American Statistical Association*.

Yousef, Q. M. and Alshaer, Y. A. (2017) 'Dragonfly Estimator : A Hybrid Software Projects ' Efforts Estimation Model using Artificial Neu ra l Network and Dragonfly Algorithm', 17(9), pp. 108–120.

Zatarain Cabada, R., Rodriguez Rangel, H., Barron Estrada, M. L. and Cardenas

Lopez, H. M. (2020) ‘Hyperparameter optimization in CNN for learning-centered emotion recognition for intelligent tutoring systems’, *Soft Computing*. Springer Berlin Heidelberg, 24(10), pp. 7593–7602.

Zhang, X., Shi, Z., Liu, X. and Li, X. (2018) ‘A Hybrid Feature Selection Algorithm For Classification Unbalanced Data Processsing’, *2018 IEEE International Conference on Smart Internet of Things (SmartIoT)*. IEEE, pp. 269–275.

Zhang, Z. (2016) ‘Multiple imputation with multivariate imputation by chained equation (MICE) package’, *Annals of Translational Medicine*, 4(2).

Zhong, S., Zhang, K., Bagheri, M., Burken, J. G., Gu, A., Li, B., Ma, X., Marrone, B. L., Ren, Z. J., Schrier, J., Shi, W., Tan, H., Wang, T., Wang, X., Wong, B. M., Xiao, X., Yu, X., Zhu, J. J. and Zhang, H. (2021) ‘Machine Learning: New Ideas and Tools in Environmental Science and Engineering’, *Environmental Science and Technology*, 55(19), pp. 12741–12754.

Ziauddin, Tipu, S. K., Zaman, K. and Zia, S. (2012) ‘Software Cost Estimation Using Soft Computing Techniques’, *Advances in Information Technology and Management (AITM)*, 2(1), pp. 233–238.

LIST OF PUBLICATIONS

Indexed Journal with Impact Factors

1. Siti Hajar Arbain et al., (2018). Combined Multiple Neural Networks and Genetic Algorithm with Missing Data Treatment: Case study of Water Level Forecasting in Dungun River – Malaysia. IAENG International Journal of Computer Science, Vol 45, no 2, pp246-254.

Indexed Conference Proceedings

1. Siti Hajar Arbain et al 2019 Adoption of Machine Learning Techniques in Software Effort Estimation: An OverviewIOP Conf. Series.: Materials. Sci. Eng.Vol(551) number 1, (012074). Conference paper indexed by scopus.
2. Siti Hajar Arbain et al 2021 Parameter Design for GMDh using Taguchi in Software Effort Estimation IOP Conf. Series.: Materials. Sci. Eng.Vol(551) number 1, (012074). Conference paper indexed by scopus.
3. "Adoption of Heuristic Techniues in Software Effort Estimation", Presenter of Postgraduate Annual Research Seminar (PARS 2018).
4. "Missing Data Handling Preparation for Software Effort Estimation" Presenter of Postgraduate Annual Research Seminar (PARS 2019).

   .

Research Grant Proposal

1. Research University Grant Scheme, Tier 2 Research Proposal, PY/2017/02027, Q.J130000.2628.15J16 Research Grant Proposal Accepted (Feb 2018)